UNITED STATES INTERNATIONAL TRADE COMMISSION
WASHINGTON, D.C.
Before Sidney Harris
Administrative Law Judge

In the matter of:	)	
	)	
Certain Garage Door	)	**Inv. No. 337-TA-459**
Operators Including	)	
Components Thereof	)	

OFFICE OF THE SECRETARY
US INTERNATIONAL TRADE COMMISSION
0067
01 NOV 13 P4:37

RESPONDENTS LYNX INDUSTRIES, INC., NAPOLEON SPRING WORKS, INC., AND GUARDIAN ACCESS CORPORATION'S MEMORANDUM OPPOSING CHAMBERLAIN'S CROSS MOTION FOR SUMMARY DETERMINATION

Respondents Lynx Industries, Inc. Napoleon Spring Works, Inc., and Guardian Access Corporation (collectively "Napoleon") oppose Complainant Chamberlain's cross motion for summary determination alleging infringement of the '364 and '703 patents because the accused Guardian GDOs at a minimum lack a claimed function of the Memory Selection Switch, that being, store each ***newly*** received code in a memory location that is ***different*** from the memory location last used to store the previously received code. The accused GDOs ***always*** store newly received codes in the ***same*** memory location. In addition, the accused GDOs do not have multiple transmitters or a decoder.

Chamberlain's allegations of infringement cannot stand because its proposed claim construction is completely unsupported by the intrinsic evidence. To arrive at its claim construction, Chamberlain wholly ignored the patent specification and file history, and instead improperly based its construction on extrinsic evidence, such as dictionary definitions, textbooks and witness testimony. Because it is fundamentally improper to rely on extrinsic evidence when intrinsic evidence alone will resolve any ambiguity in a disputed claim element, Chamberlain's

This is public rec party. See letter dated 11-14-01

construction cannot be sustained. *Vitronics Corp. v. Conceptronic, Inc.*, 90 F.3d 1573, 1576 (Fed. Cir. 1996).

Chamberain's theory of infringement depends on a claim interpretation that ignores the claimed requirement that each newly received transmitter code must be stored in a different memory location that is different from the memory location used to store the previous code. In addition, Chamberlain contends that the claimed Memory Selection Switch has the ability to move previously stored codes around to different memory locations during the program mode, despite the fact that the patents-in-suit make absolutely no mention of such a shuffling functionality. Indeed, the common patent specification of the '364 and '703 patents discloses only a single methodology for storing ***newly*** received transmitter codes described above. Moreover, the patent specification and file history clearly and unambiguously define "code" as "the ***new*** transmitter identity code that is to be programmed into the GDO." (Exhibit I at LX 1485-86)[1]. There is absolutely no teaching whatsoever that "code" refers to transmitter codes that have been previously stored in memory. More importantly, neither the patent specification, nor the claims, relate to moving previously stored codes around to different locations within the memory. Therefore, it is impossible to construe the Memory Selection Switch claim element, which is common to all the patent claims, to operate in the same manner as the code storage function of the accused Guardian GDOs. Furthermore, the Guardian devices do not have a "decoder" as required by the claims, and there is a disputed issue of fact whether the accused devices include multiple transmitters.

For these reasons, Chamberlain's cross motion should be denied.

[1] The exhibits referenced in this opposition brief are, in some cases, the same exhibits used to support Napoleon's motion for summary determination. Additional copies are attached for the Court's convenience.

I. CHAMBERLAIN CANNOT EXPAND THE CLAIMS BEYOND WHAT IS DISCLOSED IN THE PATENT SPECIFICATION

The law prohibits Chamberlain's attempt to expand the limitations of the claims beyond the written description in the patent specification, let alone adopt a construction that is contrary to the arguments it made to the Patent Examiner during the prosecution of the patent application. The specification of a patent acts as a dictionary when it expressly defines terms used in the claims or when it defines terms by implication. *Bell Atlantic Network Serv. v. Covad Communications Group,* 262 F.3d 1258, 1268 (Fed. Cir. 2001). Thus, the specification is always highly relevant to the claim construction analysis. *Id.* Usually the specification is dispositive of claim construction; it is the single best guide to the meaning of a disputed term. *Id.* Moreover, the preferred embodiments in the patent specification "can provide guidance as to the meaning of the claims, thereby dictating the manner in which the claims are to be construed, even if the guidance is not provided in explicit definitional format." *Id.* (citing *SciMed Life Sys., Inc. v. Advanced Cardiovascular Sys., Inc.*, 242 F.3d 1337, 1344 (Fed. Cir. 2001)). In addition, the prosecution history should be consulted to determine whether there were any express representations made in obtaining the patent regarding the scope and meaning of the claims. *Id.* Extrinsic evidence, on the other hand, may ***not*** be used to vary, contradict, expand or limit the claim language from how it is defined, even by implication, in the specification or file history. *Id.* at 1269.

The recently decided *Bell Atlantic* case is particularly on point. The patent owner in *Bell Atlantic* tried to expand the meaning of the claim term "mode" beyond the three broad categories described in the patent specification in order to read the claims on the accused device. The district court had granted summary judgment of no infringement and the patent owner appealed, arguing that the court had improperly limited the claims to the preferred embodiments disclosed

in the specification. The Federal Circuit disagreed and affirmed the district court, finding that the term "mode" was indeed limited to what was disclosed in the patent specification. The Court stated that,

> the written description 'can provide guidance as to the meaning of the claims, thereby dictating the manner in which the claims are to be construed, even if the guidance is not provided in explicit definitional format.' ***Thus, when a patentee uses a claim term throughout the entire patent specification, in a manner consistent with only a single meaning, he has defined that term 'by implication.'***

Id. at 1270 (emphasis added). Because the patent owner used the term "mode" consistently and provided no alternative meanings, the Federal Circuit refused to expand the meaning beyond what was disclosed in the patent specification.

In the present case the Chamberlain patents disclose only a ***single*** embodiment of the *software* Memory Selection Switch, and more importantly, Chamberlain's inventors provided no alternative means to accomplish the required functions of the Memory Selection Switch. Accordingly, there can only be one meaning attributed to the *software* Memory Selection Switch, and that meaning is found in the written patent specification, which specifically incorporates the flow diagram of FIG. 3. The correct claim construction for the *software* Memory Selection Switch must comport with the flow diagram of FIG.3 and, at a minimum, must include,

> *a software switch with different positions that functions to store a first transmitter code in a first memory storage location and that functions to store a second transmitter code in a memory storage location that is different from the first storage location.*

As described more fully below, Chamberlain's proposed claim construction is legally wrong, primarily because it ignores the intrinsic evidence, it relies wholly on contradictory extrinsic evidence, it improperly expands the scope defined by the patent specification by ignoring required functions, and it specifically ignores the requirement that each newly received

transmitter code be stored in a memory location different from the location used to store the last received code.

II. CHAMBERLAIN'S CLAIM CONSTRUCTION IS FLAWED

Chamberlain's basis for asserting infringement is based on a legally flawed construction of the Memory Selection Switch element. In order for Chamberlain to read its patent claims on the accused Guardian GDOs, the claim term "code" must have a meaning that is completely contrary to that stated in the patent specification, the file history and the meaning set forth by the Federal Circuit in *Overhead Door Corp. v. Chamberlain Group, Inc.*, 194 F.3d 1261 (Fed. Cir. 1999)(Exhibit E). Likewise, Chamberlain has improperly dissected the flow diagram of FIG. 3 of the patent specification to support its flawed claim construction without citing any evidentiary authority. Finally, Chamberlain has failed to show that the accused Guardian GDOs have a code location pointer that increments by a value of "one" until a maximum value of five is reached, and then resets to "one."

A. "Code" Means *Newly* Received Codes – Not Previously Stored Codes

Chamberlain has improperly expanded the meaning of the term "code" to include transmitter codes that have already been stored in memory. According to Chamberlain's theory, the Memory Selection Switch not only stores newly received transmitter codes, but also has the newfound ability to move or shuffle each previously stored code to different location within the memory. Chamberlain conveniently ignores that the patent only provides that each newly received code is always stored in a different memory location, i.e. memory locations 1, 2, 3, 4, or 5. Likewise, Chamberlain fails to cite to any intrinsic evidence to support such a newfound code shuffling function; instead it offers only improper extrinsic evidence of Mr. Rhyne. His recently prepared declaration is clearly influenced by Napoleon's non-infringement arguments, coupled

with the need to expand the scope of the claims in order to cover Guardian's accused GDOs. Moreover, Chamberlain's unsupported claim construction is contrary to the meaning of the term "code" found in the patent specification.

The patent specification nowhere teaches that previously stored codes are moved to different locations during the learning operation (program mode), or for that matter, at any time during the operation of the claimed GDO system. The word "code" is only used to refer to ***newly*** received transmitter codes. For example, the patent specification states,

> . . .wherein the receiver can be placed into a program mode wherein it will receive and store two or more codes corresponding to two different transmitters (Col. 1, lines 47-49).
>
> In the program mode a code must be received four times in a row in order to be permanently stored in the receiver (Col. 2, lines 17-19 and Col. 4, lines 34-37
>
> The switch 23 can be moved to a number of selected positions . . . to allow the particular code of a number of transmitters to be stored in the receiver . . .(Col. 3, lines 14-17).
>
> Thus, the codes of the transmitters 26 and 28 are transmitted and stored in the memory address storage 47 during the program mode . . .(Col. 4, lines 6-8).
>
> If switch 22 is in the program mode as shown in FIG. 3 when an incoming signal from a transmitter is received, the flow diagram is followed so as to store the new incoming [code] in the code location pointed to by the code location pointer 23 (Col. 4, lines 57-61).
>
> It is seen that the present invention allows a receiving system to respond to one of a plurality of transmitters which have different unique codes which can be stored in the receiver during a program mode. Each time the "program mode switch" 22 is moved to the program position, a ***different*** storage area as determined by the switch 23 can be connected so that the new transmitter code would be stored in that address (Col. 4-5, lines 63-67 and lines 1-3)(emphasis added).

Each of the above excerpts from the patent specification makes it clear that the inventors used the word "code" in the claims to mean only ***newly*** received transmitter codes and ***not*** codes

already stored in memory. Indeed, the inventors distinguished newly received codes from previously stored codes when they referred to the latter as "old codes." (Col. 5, line 4). Significantly, the term "old codes" is ***not*** used anywhere in the any of the asserted claims.

The patent prosecution history supports this claim construction. During the prosecution of the '703 patent, Chamberlain submitted a sworn declaration as part of its efforts to convince the Patent Examiner that the pending claims were patentable. The declarant was none other than Mr. Thomas Rhyne. In that declaration, Mr. Rhyne told the Examiner the following:

- "Code" refers to the ***new*** transmitter identity code that is to be programmed into the GDO.
- . . . the "code location pointer" serves to identify the location within the memory . . .at which the ***next*** transmitter identity code to be learned is to be stored.
- . . .the microprocessor . . .uses the code location pointer . . .to determine the location . . . at which each ***newly*** learned transmitter identity code is to be stored.

(Exhibit I at LX 1485-86). There is no ambiguity in what Mr. Rhyne told the Patent Office and there can be no doubt that Chamberlain used the term "code" consistently through out the patent specification to mean ***newly*** received codes. Nowhere in the prosecution is there any teaching that "code" or "code location pointer" refers to anything but newly received transmitter codes. Now, to support its allegations, Chamberlain has had to resort to a complete fabrication of the meaning of the term "code" to justify its contention that the claims read on the accused Guardian GDOs.

The Federal Circuit opinion in *Overhead Door v. Chamberlain*, upon which Chamberlain so heavily relies, supports Napoleon's, not Chamberlain's, construction of the claim element "code." The Federal Circuit squarely addressed the meaning of the claim term "code" in its construction of the claim element "different codes."

> The claim language requires association of each code with a different transmitter. The written description notes that each code uniquely identifies a transmitter.
>
> The language of both claims recite "that the code of said first transmitter will be stored in said memory means." Properly construed, this language requires the memory to retain the code it associates with the "first transmitter."
>
> This reading of the claim language finds support in the written description, which explains that a code is associated with a transmitter.

(Exhibit E, 194 F.3d at 1275). Like the patent specification and file history, there is nothing in the Federal Circuit opinion that supports Chamberlain's unsupported claim construction that the word "code" refers to previously stored transmitter codes, when in reality it refers only to ***newly*** received codes. The intrinsic evidence simply does not support Chamberlain's claim construction.

B. Chamberlain Has Improperly Parsed FIG. 3

Central to Chamberlain's proposed construction of the *software* Memory Selection Switch claim element is its wrongful attempt to convince the Court to ignore 80% of the decision boxes of the flow diagram depicted in FIG. 3 of the patent specification. If the Court accepts Chamberlain's construction, then four of the six stated functions of the *software* Memory Selection Switch must likewise be ignored. Indeed, in its brief at page 13, Chamberlain contends that only two of the eighteen decision boxes of FIG. 3 need be considered to construe the Memory Selection Switch. Further, Chamberlain brazenly strikes out half the functions of the second box, contending that the Court can ignore the requirement that the pointer value be checked and, if over "five," reset to a value of "one." Chamberlain has offered absolutely no support for its tortured parsing of FIG. 3. Moreover, Chamberlain's construction is completely contrary to the intrinsic evidence, specifically what Mr. Rhyne told the Patent Examiner years ago during the prosecution of the application.

> The actions of this software memory selection switch are ***fully*** described in Figures 3 and 4 . . . as well as at Column 4, Lines 31 through 68 . . .

(Exhibit I at LX 1485-86)(emphasis added). Nowhere in the file history or the patent is there any mention that the Memory Selection Switch will work using the truncated FIG.3 now proposed by Chamberlain. Chamberlain ignores all the intrinsic evidence and relies instead on the recently prepared, litigation-induced extrinsic testimony of Mr. Rhyne. This improper extrinsic evidence of Mr. Rhyne must be distinguished from Mr. Rhyne's sworn declaration that he submitted to the Patent Office years ago during the prosecution of the application. That earlier Rhyne declaration is intrinsic evidence and should be considered to properly construe the claim terms.

In addition to the intrinsic evidence, the Federal Circuit also indicated that FIG. 3 as a whole describes the *software* Memory Selection Switch. The Federal Circuit could not have been more clear: "Figure 3 of the '364 patent, shown below, illustrates how the invention receives and validates codes. " (Exhibit E, 194 F.3d at 1268). Moreover, the Court relied on the language of the patent specification in construing the Memory Selection Switch:

> If the switch 22 is in the 'program' mode as shown in FIG.3 when the incoming signal from a transmitter is received, ***the flow diagram is followed*** so as to store the new incoming [code] in the code location . . .

Id. (emphasis added). The Court further stated that "'[s]witch means,' when properly construed, also covers the software-based embodiment described in Figure 3." *Id.* at 1273. Accordingly, FIG. 3 must be viewed as a whole, and not selectively dissected as Chamberlain urges in order to read the claims on the accused Guardian devices. An examination of FIG. 3 shows that the software routine that begins the Memory Selection Switch begins at the top right side of the flow diagram where the label (A) is shown. It does not, as Chamberlain asserts, start in the lower right hand corner.

Moreover, there is no other embodiment in the patent that describes or suggests operation of the software version of the Memory Selection Switch other than the complete flow diagram of FIG. 3. Further, there is absolutely no support in the specification or prosecution history to suggest that the software routine can begin where Chamberlain proposes. Accepting Chamberlain's starting point would completely eliminate at least six separate functions required by the software flow diagram when the GDO is in the program mode. These functions must be performed each and every time a new code is received from a transmitter. Those six functions are outlined below:

1) validating newly received transmitter codes four times and incrementing a counter;
2) checking each newly received code against previously stored codes;
3) selecting a memory location for the newly received code;
4) incrementing a code location pointer;
5) checking the value of the incremented code location pointer; and
6) resetting the value of the incremented code location pointer.

Indeed, the written portion of the patent specification specifically sets forth that when in the program mode,

> . . . a code must be received four times in a row in order to be permanently stored in the receiver.

(Col. 2, lines 17-19 and Col. 4, lines 34-37). The decision boxes on the top right of FIG. 3 and those on the left side of FIG. 3, perform this function.

Chamberlain's infringement contentions fail because none of the required functions are performed by the accused Guardian GDOs. Napoleon's technical expert, Mr. Gafford, concluded the following after his review of the Guardian GDOs.

> 21. There is no indexing of a "pointer" to direct a newly received code to a particular memory location. Because there is no indexing, there is no checking the value of the pointer to determine whether it is equal to 5. There is no

resetting of a pointer if the value equals 5. And, there is no erasing of all codes in memory when a pointer is reset.

22. When in program mode, none of the software on the [Guardian] product checks the newly received code to see if it already matches a code already stored. As such, there is no repetitive checking of the stored codes. When a new code is received and the unit is in program mode, the branch instruction at line 180 of door.asm jumps to STORE_PASS where the new code is stored in locations 00 and 01 without regard to the other code contents of memory.

23. Since there is no checking function to determine whether a new code entered during program mode matches an existing stored code, the same code can be entered 6 times. Therefore, it is possible to fill all available memory locations with the same code.

(Exhibit G). Without those functions there is no memory Selection Switch and there can be no finding of infringement, either literally or under the doctrine of equivalents. *Pennwalt Corp. v. Durand-Wayland, Inc.*, 833 F.2d 931 (Fed. Cir. 1987).

C. Guardian's GDOs Do Not Increment

Chamberlain's motion for summary determination should also be denied because it fails to establish that the Guardian GDOs perform the incrementing step required by the claimed Memory Selection Switch. Even assuming Chamberlain's improperly dissected flow diagram of FIG. 3 is a correct representation of the Memory Selection Switch, there is no dispute that FIG. 3 requires that the "code location pointer" must increment after storing a first newly received transmitter code so that it points to a ***different*** memory address to store the second or next received code. (see Chamberlain's brief at pages 13 -14). "The code location pointer is automatically incremented by the microprocessor each time the receiver learns a new transmitter code (last action block on right)." *Id.* The patent specification, including FIG. 3, makes clear that this function is a step function, where, after each newly received code is received and stored in memory, the code location pointer is increased (incremented) by a value of "one" until each memory location (i.e. 1, 2, 3, 4, and 5) is filled. Once all memory locations are filled, the value

of the code location is reset to "one." This is confirmed by the patent file history, where Chamberlain, through the sworn declaration of Mr. Rhyne, told the Patent Examiner,

> [a]s the software loop disclosed in Figure 3 is executed by the microprocessor unit 44, ***the value of the code location pointer will be moved from one address to the next until all five addresses in memory that were set aside for transmitter identity codes have been used to store learned identity codes.*** The specific disclosure of this loop is (a) "Increment Code Location Pointer", followed by (b) "If Pointer Increments Over Five Then Load Code Location Pointer With One." Step (a) is the "body" of the loop; ***it uses the "increment" operational to increase the value of the code location pointer by one.***

(Exhibit I at LX 1573)(emphasis added). Apparently, Chamberlain and Mr. Rhyne, have forgotten what they told the Patent Office because they now ignore the very limitation they relied on to obtain the patent. Not surprisingly, in Mr. Rhyne's recent declaration describing the operation of the Guardian GDOs he is unable to identify the "increment" functionality required by the claims. This is because there is no such function in the accused Guardian devices. The Guardian GDOs ***always*** store each newly received transmitter code in the same memory location (0/1). (Exhibit G). Since there is no "code location pointer" to increment by "one," there can be no checking function see if the value is "over five." *Id.* And, likewise there can be no resetting the pointer to "one." *Id.*

D. The "software controlled memory selector" Is *Not* A New Claim Element

Chamberlain's argument that the "software controlled memory selector" should have a different meaning and scope than the other claim terms used to describe the *software* Memory Selection Switch is ridiculous. The term "software controlled memory selector" is not used or mentioned anywhere in the patent specification. The only remote reference is the ***one*** instance in the specification that refers to "the selector switch 23." (Col. 4, line14). Switch 23, as shown in FIG. 2, is none other than the *mechanical* Memory Selection Switch. There is absolutely no

intrinsic evidentiary support for Chamberlain's contention that the "software controlled memory selector" is anything but the same Memory Selection Switch that is common to all the asserted claims. Any other conclusion runs afoul of 35 U.S.C. § 112. A phrase added to a claim by amendment cannot enlarge the scope of the claim beyond that supported in the specification and cannot change the disclosure in a way contrary to its substance as filed. *Tandon Corp. v. United States ITC*, 831 F.2d 1017 (Fed. Cir. 1987).

III. GUARDIAN'S GDOS DO *NOT* HAVE MULTIPLE TRANSMITTERS

Another reason to deny Chamberlain's motion is that there is an issue of fact whether the accused Guardian GDO system have multiple transmitters as required by the calims. Chamberlain selectively relied on the equivocal deposition testimony of Napoleon's owner, Mr. Robert Schram, in an attempt to satisfy the claim limitation that requires the GDO system to have a "plurality" of radio transmitters. What Chamberlain did not tell the Court was that it had in its possession ***unequivocal*** testimony, taken ***before*** Mr. Schram's deposition, indicating that the accused Guardian GDOs are sold with only a ***single*** transmitter. Chamberlain learned the true facts from Mr. Mark Schram who is directly responsible for purchasing of the Guardian GDOs for Lynx, the parent company of Napoleon. (Exhibit N, pg. 13) Chamberlain ignored Mr. Mark Schram's deposition testimony and instead relies on the deposition of Mr. Robert Schram, who was unclear whether or not his company sold one or more transmitters with each GDO system. Mr. Mark Schram's testimony, on the other hand, was unequivocal.

> Q. And Guardian Access is the supplier for the entire garage door openers?
> A. Yes.
> Q. And they are bought from Guardian Access directly by Lynx; is that right?
> A. Yes.
> Q. And when they are bought, are they bought as complete garage door opener units, including the remote control?

A. Each garage door opener that we purchase from Guardian ***comes with one remote control*** or one transmitter.
Q. They are sold as a package together?
A. Yes.
Q. Do you sell those packages together when you sell them to your customers?
A. Yes, ***one garage door opener is sold with one transmitter.*** They come as a package.

(Exhibit N, pgs. 69-70)(emphasis added).

After learning of the allegations made in Chamberlain's motion, Mr. Robert Schram personally conducted an investigation of Napoleon's corporate records and has confirmed the testimony of Mr. Mark Schram..

> I personally performed an investigation of the corporate records of Napoleon and determined Napoleon has at all times purchased prepackaged GDO systems from Guardian that include one receiver and only one transmitter in a single box. At no time has Napoleon ever purchased a prepackaged GDO system from Guardian that had more than one transmitter. Moreover, Napoleon has never altered the prepackaged GDO system to include additional transmitters. Based on the results of my investigation and based upon my review of my deposition testimony, it is clear that I was mistaken when I testified that Napoleon may have tried to sell a prepackaged GDO system with more than one transmitter. The fact is that Napoleon has never sold a prepackaged GDO system with more than one transmitter.

(Exhibit O, ¶ 5).

There is no question that the claims of the patents-in-suit require more than a single transmitter; otherwise the benefit of storing multiple transmitter codes cannot be satisfied. Moreover, all the claims of the patents-in-suit are written as system-type claims covering a collection of physical devices (i.e. transmitters, receiver, switches, decoder, microprocessor, etc.), each of which performs specific functions to ensure operation of the entire system. In order to find infringement, each of the claimed devices must be found in the accused product. Because Napoleon has ***never*** sold a GDO system with more than ***one*** transmitter, there can be no

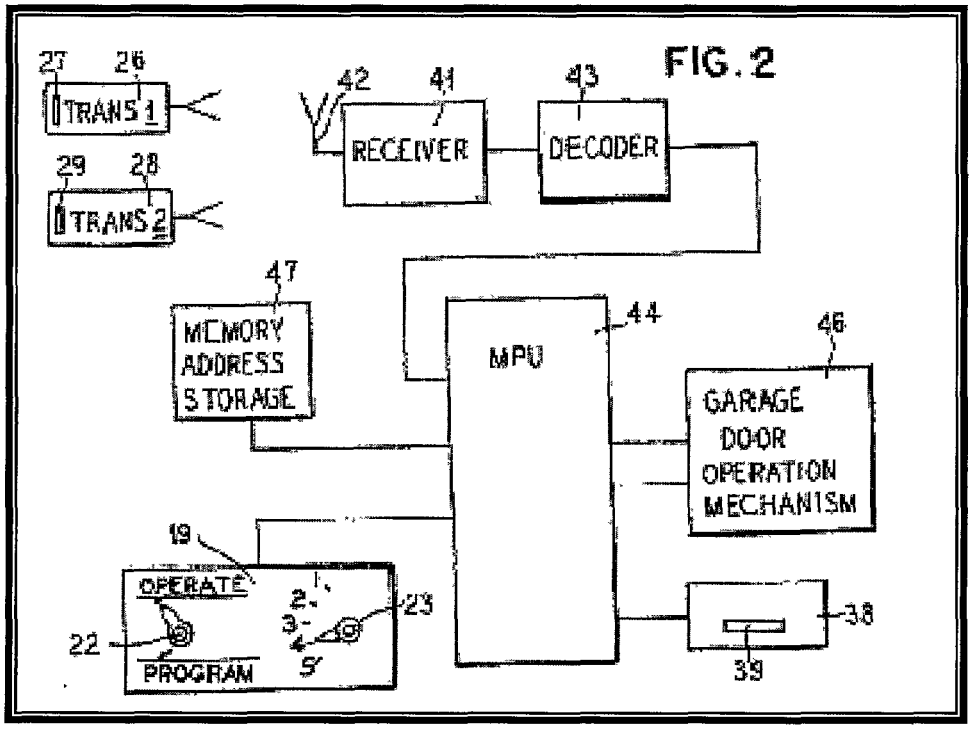
direct infringement of the claims. At a minimum there is a genuine issue of fact as to whether the accused devices have the required "multiple" transmitters.

IV. GUARDIAN'S GDOS DO *NOT* HAVE A DECODER

Another structural element required by each of the asserted system claims is a "decoder." The patent specification describes only one embodiment of the "decoder" and does not disclose or suggest any other acceptable alternatives. In fact, the patent specification refers to FIG.2 when it describes the operation of the decoder.

> As illustrated in FIG. 2 the garage door operator includes a receiver 41 which has a suitable antenna 42 for receiving radio frequency transmissions from the transmitters 26 and 28 and supplies an input to a decoder 43 which provides an output to a microprocessor unit 44.

(Exhibit C, col. 3, lines 1-5) As shown below, FIG. 2 of the patent specification supports the above-stated written description because it clearly shows that "decoder" 43 is a stand-alone device that is directly connected to "receiver" 42 and "MPU" 44.



No other description of the "decoder" is found in the patent specification. Using FIG. 2 for claim interpretation is proper because the Federal Circuit in *Overhead Door v. Chamberlain* likewise used FIG. 2 to interpret the meaning of the claim term "switch." The Court stated that,

> [T]o interpret the term "switch" consistently in the claim and to ***harmonize the drawing depiction [FIG. 2] with the claim language***, this Court confirms the district Court's reading of the term "switch."

(Exhibit E, 194 F.3d at 1267). This same "harmonization" requirement equally applies to the term "decoder," and requires that it be construed as a separate device that receives input from the "receiver," performs a decoding function and sends its output to a microprocessor (i.e. the "MPU").

Napoleon's technical expert, Mr. Gafford, has reviewed the the intrinsic evidence and concluded that one skilled in the art would construe the claim term "decoder" to mean the following:

> The "decoder" is a separate electrical device that is electrically connected to a receiver and a microprocessor. The function of the "decoder" is to receive coded transmitter signals from the receiver. The "decoder" then decodes the transmitter signals and sends the decoded signals to a microprocessor for further processing.

(Exhibit P – Para. 7 Gafford's Second Supplemental Decl.). Following the reasoning in *Bell Atlantic*, Chamberlain has consistently used the term "decoder" throughout the patent specification and has provided no alternatives, thus no greater breath than what was originally disclosed by Chamberlain can be accorded. *Bell Atlantic Network Serv. v. Covad Communications Group,* 262 F.3d 1258, 1271 (Fed. Cir. 2001).

Mr. Gafford's study of the accused Guardian GDO reveals that it does not have a "decoder" as that term is used in the asserted claims.

> 8. The Guardian GDO, specifically the controller does not include a separate and independent decoder device such as the decoder illustrated in Figure 2 of the '364 and the '703 patents. Rather, when the Guardian GDO receives a transmitted code, the transmitted code is received by a radio frequency demodulator. The output of this demodulator is directly connected to a CPU, not a separate and independent decoder device. There is, therefore, no separate and independent decoder unit connected between a receiver and a microprocessor as those terms were used by the inventors of the '364 and the '703 patent.

(Exhibit P - Gafford's second supplemental declaration). Because the Guardian GDOs do not have a "decoder" there can be no literal infringement. Although Chamberlain did not assert infringement under the doctrine of equivalents, there could be no such finding because there is a substantial difference between the claimed "decoder" as a separate device connected to a receiver and MPU, and Guardian's software subroutine program that is integral to its microcontroller.

V. CONCLUSION

Chamberlain's allegation of infringement fails because its proposed claim construction is flawed. The intrinsic evidence does not support Chamberlain's attempt to expand the scope of the claims with respect to the *software* Memory Selection Switch, decoder, and multiple transmitter elements. The Chamberlain patent specification discloses a single embodiment of the *software* memory Selection Switch and makes absolutely no mention that alternative functions are possible. Therefore it is improper to construe the disputed claim terms to have a meaning any broader than that supported by the intrinsic evidence.

Chamberlain's patents require a *software* Memory Selection Switch that performs a number of functions, including directing newly received transmitter codes into a memory location that is ***different*** from the location where the last received code was stored. The Guardian GDO does not have the claimed *software* Memory Selection Switch because the Guardian GDO ***always*** stores each new transmitter code in the ***same*** memory location. Further,

the Guardian GDO does not perform any of the many functions required by the *software* Memory Selection Switch. Additionally, the Guardian GDOs are sold with a single transmitter and do not contain a "decoder." For these reasons, there can be no finding of literal infringement or infringement under the doctrine of equivalents. Therefore, Chamberlain's motion for summary determination should be denied.

Respectfully submitted,

Date: November 17, 2001

By: _____

David M. Frischkorn
A. Blair Hughes
Thomas E. Wettermann
Jeremy E. Noe
McDonnell Boehnen Hulbert & Berghoff
300 South Wacker Drive
Chicago, Illinois 60606
Telephone: 312/913-0001
Facsimile: 312/913-0002

George M. Sirilla
Arthur Wineburg
D. James Pak
Pillsbury Winthrop LLP
1600 Tyson Boulevard
McLean, VA 22102
Telephone: 703/905-2000
Facsimile: 703/905-2500

Attorneys for Respondents
LYNX INDUSTRIES, INC.,
NAPOLEON SPRING WORKS, INC.
GUARDIAN ACCESS CORP.

CERTIFICATE OF SERVICE

I hereby certify that a copy of the Foregoing **Respondents Lynx Industries, Inc., Napoleon Spring Works, Inc., & Guardian Access Corporation's Memorandum Opposing Chamberlain's Cross-Motion for Summary Determination** has been served on this 13th day of November 2001, as indicated, on the following:

VIA COURIER

Hon. Sidney Harris (2 copies)
Administrative Law Judge
U.S. International Trade Commission
500 E Street, S.W. - Room 317-G
Washington, DC 20436

Ms. Donna R. Koehnke (original & 6 copies)
Secretary
U.S. International Trade Commission
500 E Street, S.W. - Room 112
Washington, DC 20436

David O. Lloyd, Esq. (1 copy)
Office of Unfair Import Investigations
500 E Street, S.W. - Room 401-M
Washington, DC 20436

Thomas M. Schaumberg, Esq.
Michael L. Doane, Esq. (1 copy)
ADDUCI, MASTRIANI & SCHAUMBERG, LLP
1200 17th Street, N.W. – 5th Floor
Washington, DC 20036-3006
Counsel for Complainant

VIA FIRST CLASS U.S. MAIL

John F. Flannery, Esq. (1 copy)
Karl R. Fink, Esq.
FITCH, EVEN, TABIN & FLANNERY
120 South LaSalle Street – Suite 1600
Chicago, IL 60603
Counsel for Complainant

Ramon R. Hoch, Esq. (1 copy)
FITCH, EVEN, TABIN & FLANNERY
1801 K Street, N.W. – Ste. 401L
Washington, DC 20006-1201
Counsel for Complainant

Philippe M. Bruno, Esq. (1 copy)
DORSEY & WHITNEY LLP
1001 Pennsylvania Avenue, N.W.
Suite 300 South
Washington, DC 20004
Counsel for Respondent Wayne-Dalton Corporation

Ray L. Weber, Esq. (1 copy)
Phillip L. Kenner, Esq.
RENNER, KENNER, GRIEVE, BOBAK, TAYLOR & WEBER
First National Tower, Fourth Floor
Akron, OH 44308-1456
Counsel for Respondent Wayne-Dalton Corporation

W. Joseph Melnik, Esq. (1 copy)
Raymond A. Miller, Esq.
BENESCH, FRIEDLANDER, COPLAN & ARONOFF, LLP
2300 BP Tower
200 Public Square
Cleveland, OH 44114-2378
Counsel for Respondent Innovative Home Products, Inc.

By: _____
Jennifer L. Pugh-Nolan

US00RE35364E

United States Patent [19]

Heitschel et al.

[11] E Patent Number: Re. 35,364

[45] Reissued Date of Patent: Oct. 29, 1996

[54] CODING SYSTEM FOR MULTIPLE TRANSMITTERS AND A SINGLE RECEIVER FOR A GARAGE DOOR OPENER

[75] Inventors: Carl Heitschel, Downers Grove; Colin Willmott, Buffalo Grove, Wayne Schindler, Lisle, all of Ill.

[73] Assignee: The Chamberlain Group, Inc., Elmhurst, Ill.

[21] Appl. No 425,724

[22] Filed. Apr. 20, 1995

Related U.S. Patent Documents

Reissue of:

[64] Patent No.: 4,750,118
Issued: Jun. 7, 1988
Appl No: 792,661
Filed: Oct. 29, 1985

U.S. Applications:

[63] Continuation of Ser. No 87,142, Jul. 2, 1993, abandoned, which is a continuation of Ser. No 715,006, Jun. 13, 1991, abandoned, which is a continuation of Ser. No 398,379, Aug. 24, 1989, abandoned.

[51] Int. Cl.[6] G06F 19/00; G08C 19/00; E05F 15/20

[52] U.S. Cl. 364/400; 49/25; 340/825.69; 340/825.73

[58] Field of Search 364/130, 400, 364/167.01, 825.04, 825.3, 825.31, 825.44, 825.56, 825.69, 825.71, 825.72, 825.73, 825.74, 825.75, 825.76, 539, 542, 543; 49/25, 28, 31, 70, 324; 318/16, 262–266, 282, 466–468; 361/171, 172; 455/186.1, 186.2

[56] References Cited

U.S. PATENT DOCUMENTS

Re 29,525	1/1978	Willmott	325/37
3,911,397	10/1975	Freeny, Jr.	340/147 MD
4,037,201	7/1977	Willmott	340/167 R
4,081,747	3/1978	Meyerle	325/37
4,119,896	10/1978	Estes, III et al.	318/266
4,130,738	12/1978	Sandstedt	179/90 B
4,178,549	12/1979	Ledenbach et al	325/38 R
4,263,536	4/1981	Lee et al.	318/266

(List continued on next page.)

FOREIGN PATENT DOCUMENTS

0099762A1	2/1984	European Pat. Off.
0143309A2	6/1985	European Pat. Off. .
0212050A2	3/1987	European Pat. Off. .
0319781A2	6/1989	European Pat Off. .
0401673A1	12/1990	European Pat. Off.
0561361A1	9/1993	European Pat. Off
0563517A1	10/1993	European Pat. Off
2746532A1	4/1979	Germany .
2856337	7/1980	Germany
2939589A1	4/1981	Germany
2941394A1	4/1981	Germany
55-114771	4/1980	Japan
59-80872	5/1984	Japan
1423126	9/1974	United Kingdom

OTHER PUBLICATIONS

"Garagentorantrieb mit Funkfernsteuerung," by Torsten Meyer-Staufenbiel, pp 115–117, dated 1975.

Alltronik Opposition Brief of Jul 8, 1990, corresponding to German Patent No 36 25 555

Dorma Opposition Brief of Aug. 8, 1990 of corresponding German Patent No. 36 25 555.

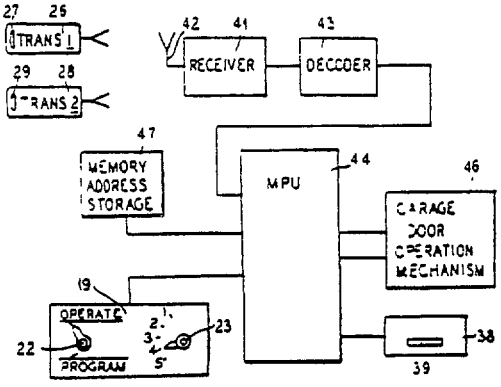
Primary Examiner—Joseph Ruggiero

Attorney, Agent, or Firm—Fitch, Even, Tabin & Flannery

[57] ABSTRACT

The present invention comprises a system for remote control of garage doors and other devices wherein an extremely large number of codes are available for remote transmitters for operating the garage operator and wherein each transmitter has its own unique and permanent nonuser changeable code. The receiver at the garage door operator is capable of storing and remembering a number of different codes corresponding to different transmitters such that the receiver can be programmed so as to actuated by more than one transmitter code thus allowing two or more transmitters to actuate the same garage door operator and wherein the receiver stores the valid codes for the different transmitters

8 Claims, 3 Drawing Sheets



LX 0001

U S PATENT DOCUMENTS

4,291,411	9/1981	Müller et al	455/33
4,305,060	12/1981	Apple et al	340/825.5
4,322,855	3/1982	Mogi et al	455/51
4,328,540	5/1982	Matsuoka et al	364/[illegible]
4,385,296	5/1983	Tsubaki et al.	340/825.72
4,418,333	11/1983	Schwarzbach et al	340/310 A
4,422,071	12/1983	DeGraff	340/825.44
4,454,509	6/1984	Buennagel et al	340/825.69
4,464,651	8/1984	Duhame	340/521
4,491,774	1/1985	Schmitz	318/282
4,509,093	4/1985	Stellberger	361/172
4,529,980	7/1985	Liotine et al	340/825.52
4,535,333	8/1985	Twardowski	340/825.69
4,550,311	10/1985	Galloway et al	340/531
4,581,606	4/1986	Mallory	340/539
4,596,985	6/1986	Bongard et al	340/825.69
4,602,357	7/1986	Yang et al	367/93
4,633,247	12/1986	Hegeler	340/825.69
4,638,433	1/1987	Schindler	364/400
4,652,860	3/1987	Weishaupt et al.	340/539
4,665,397	5/1987	Punnow	340/825.56
4,755,792	7/1988	Pezzolo et al	340/538
4,758,835	7/1988	Rathmann et al	340/825.31
4,866,434	9/1989	Keenan	340/825.72
4,878,052	10/1989	Schulze	340/825.69
4,912,463	3/1990	Li	340/825.69
4,988,992	1/1991	Heitschel et al.	340/825.69

FIG. 1

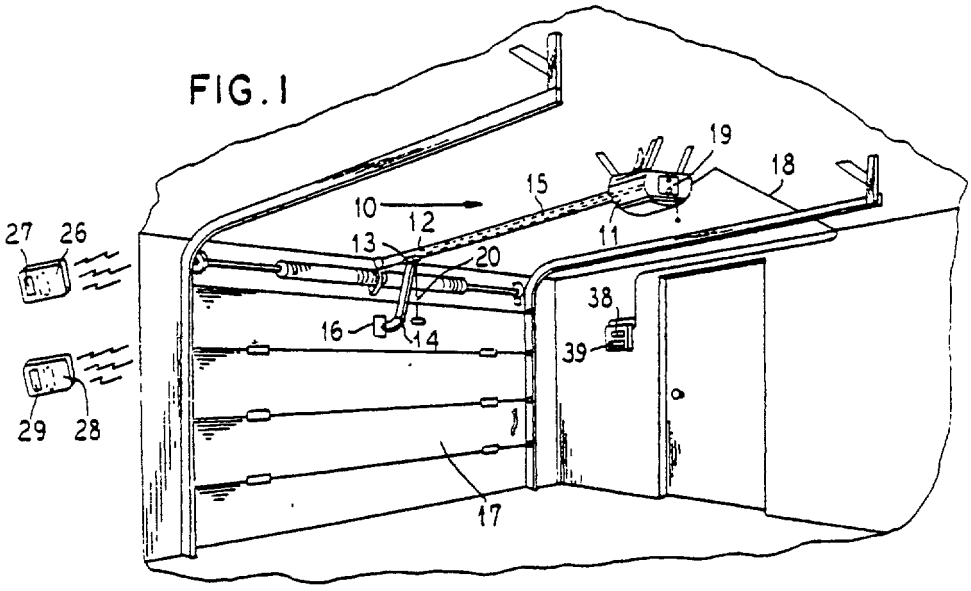

27
26
29
28
10
12
13
20
15
11
19
18
16
14
38
39
17

FIG. 2

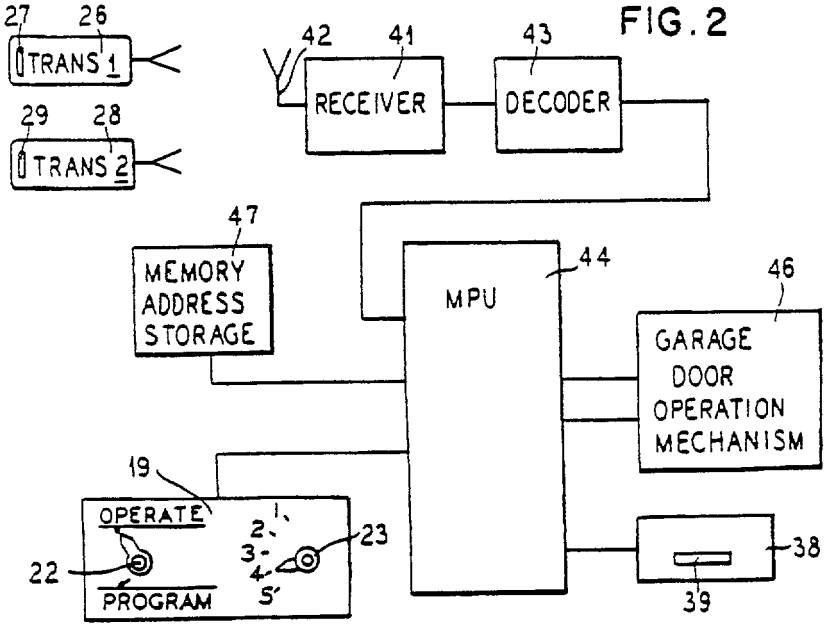
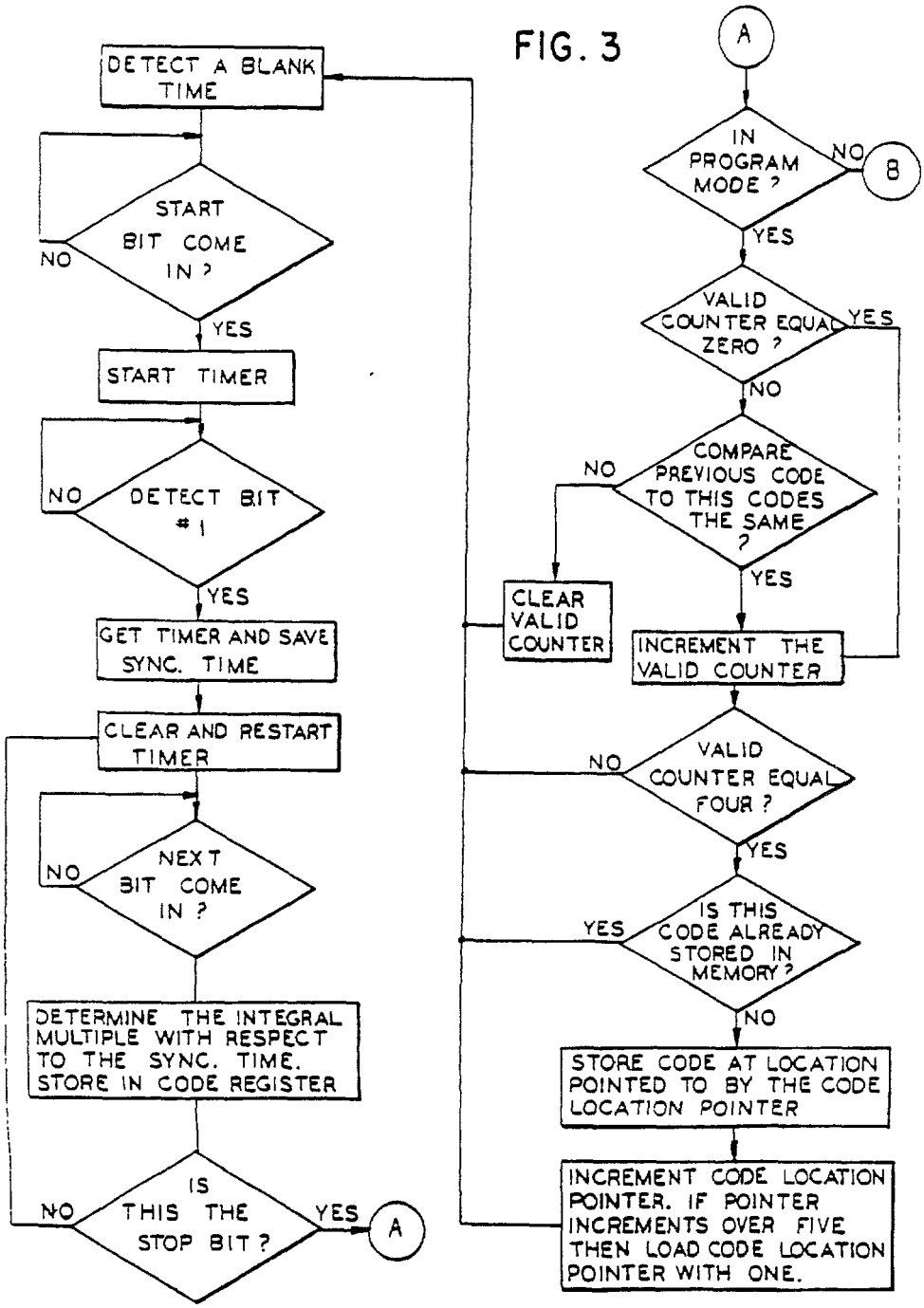

27
26
TRANS 1
29
28
TRANS 2
42
41
RECEIVER
43
DECODER
47
MEMORY ADDRESS STORAGE
MPU
44
46
GARAGE DOOR OPERATION MECHANISM
19
OPERATE
PROGRAM
22
23
1
2
3
4
5
38
39

FIG. 3


DETECT A BLANK TIME
START BIT COME IN ?
NO
YES
START TIMER
DETECT BIT #1
NO
YES
GET TIMER AND SAVE SYNC. TIME
CLEAR AND RESTART TIMER
NEXT BIT COME IN ?
NO
DETERMINE THE INTEGRAL MULTIPLE WITH RESPECT TO THE SYNC. TIME. STORE IN CODE REGISTER
IS THIS THE STOP BIT ?
NO
YES
A
A
IN PROGRAM MODE ?
NO
B
YES
VALID COUNTER EQUAL ZERO ?
YES
NO
COMPARE PREVIOUS CODE TO THIS CODES THE SAME ?
NO
YES
CLEAR VALID COUNTER
INCREMENT THE VALID COUNTER
VALID COUNTER EQUAL FOUR ?
NO
YES
IS THIS CODE ALREADY STORED IN MEMORY ?
YES
NO
STORE CODE AT LOCATION POINTED TO BY THE CODE LOCATION POINTER
INCREMENT CODE LOCATION POINTER. IF POINTER INCREMENTS OVER FIVE THEN LOAD CODE LOCATION POINTER WITH ONE.

LX 0004

FIG. 4

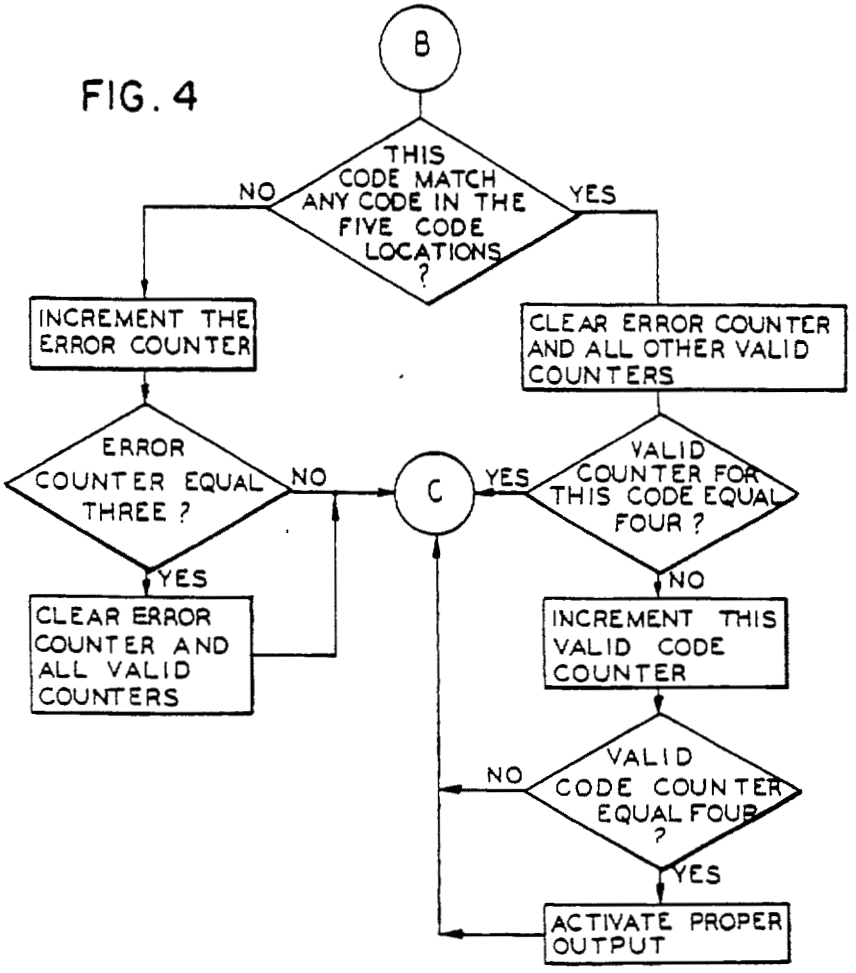
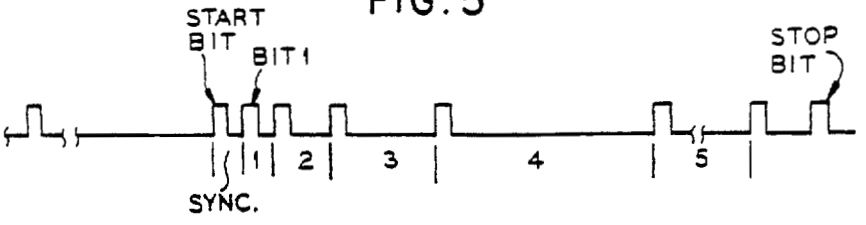


FIG. 5



LX 0005

CODING SYSTEM FOR MULTIPLE TRANSMITTERS AND A SINGLE RECEIVER FOR A GARAGE DOOR OPENER

Matter enclosed in heavy brackets [ ] appears in the original patent but forms no part of this reissue specification; matter printed in italics indicates the additions made by reissue.

This application is a continuation of application Ser. No. 087,142 filed Jul 2, 1993, abandoned, which is a continuation of application Ser. No 715,006, filed Jun. 13, 1991, abandoned, which is a continuation, of application Ser No 398,379 filed Aug. 24, 1989 abandoned.

CROSS-REFERENCES TO RELATED APPLICATIONS

This application comprises an improvement on application Ser No. 615,339, filed May 30, 1984, U.S. Pat. No. 4,638,433, in which the inventor is Wayne R. Schindler assigned to the assignee of the present application.

BACKGROUND OF THE INVENTION

1. Field of the Invention

This invention relates in general to garage door operators and in particular to a novel garage door operator wherein the receiver can be energized by two or more transmitted codes which are stored in the receiver.

2. Description of the Prior Art

Garage door operators of the prior art used transmitters in which the code can be changed by various methods as, for example, by moving two position switches to change the code. Such systems have also used code changing switches in the receiver so that the receivers can be set to correspond to the selected transmitter code.

It has also been known to use fixed frequency transmitters and fixed frequency receivers such that if the transmitted frequency matches the receiver frequency the receiver will respond.

SUMMARY OF THE INVENTION

It is an object of the present invention to provide a plurality of transmitters wherein each transmitter has its own unique and permanent non-user changeable code and wherein the receiver can be placed into a program mode wherein it will receive and store two or more codes corresponding to two different transmitters. The number of codes which can be stored in transmitters can be extremely high as, for example, greater than one million codes. Thus, the invention makes it possible to eliminate the requirements for code selection switches in the transmitters.

In the present invention the decoder module in the receiver will be capable of learning several different transmitted codes which will eliminate code switches in the receiver and also provides for multiple transmitters for actuating the garage opener.

The communication link can be various system such as radio frequency, light, wires, etc.

The invention makes it very easy for the user to operate the system and more secured code systems are available due to the higher number of available codes.

An encoded signal will be utilized wherein a pulse and blank time comprises a sync time base and different link pulses such as 1, 2, 3 or 4 milliseconds can be selected so as

to provide different codings. Each datum can be 1, 2, 3 or 4 times the length of the sync pulse. The timing is from the rising edge to rising edges of the pulse and with ten data bits the number of codes can be in excess of one million codes

In the invention, each transmitter encoder will contain a chip which contains a unique code and the receiver will be able to memorize two or more as, for example, five different transmitter codes. This eliminates the need to have coding switches in either the transmitter or receiver. This eliminates the requirement that the user set the code switches so they match since the code switches are eliminated.

In the invention, during an operate mode, a receiver code must match an already programmed code four times in order to operate the garage door. This match is referred to as a valid code. Each valid code can be separated by up to two error codes and still have the output indicated as accurate

In the program mode a code must be received four times in a row in order to be permanently stored in the receiver. Any error code will reset the valid code counter.

The advantage of the coding scheme are:

1. Higher peak power without exceeding the FCC rules which gives longer transmitter range.

2. Eliminate code switches in the transmitter and receiver making it easier for a customer to install and operate his garage door operator.

3. Customers having more than one transmitter will not have to match codes.

4 More secure codes due to the higher number of combinations which are available. Other objects, features and advantages of the invention will be readily apparent from the following description of certain preferred embodiments thereof, taken in conjunction with the accompanying drawings although variations and modifications may be effected without departing from the spirit and scope of the novel concepts of the disclosure and in which:

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a perspective view illustrating a garage door operator;

FIG. 2 illustrates in block form the invention,

FIG. 3 comprises a flow diagram;

FIG. 4 is a continuation of the flow diagram; and

FIG. 5 illustrates the coding scheme.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

FIG 1 illustrates a garage door operator **10** mounted to the ceiling of a garage and connected to operate a door **17** Garage door operator **10** has a head unit **11** which is supported from the ceiling which includes a motor which drives a suitable chain to which a trolley **13** is attached so that it moves along a rail **12**. The trolley **13** has a release cord **20** and pivotally carries a lever arm **14** which is attached to a bracket **16** mounted to the door so as to raise and open it by pulling it along conventional rails.

The header unit **11** includes a receiver and operating mechanism and can be actuated from a control unit **38** which has a operate switch **39**.

The garage door operator can also be operated by the transmitters **26** and **28** which have operate transmit buttons **27** and **29**, respectively.

As illustrated in FIG. 2 the garage door operator includes a receiver 41 which has a suitable antenna 42 for receiving radio frequency transmissions from the transmitters 26 and 28 and supplies an input to a decoder 43 which provides an output to a microprocessor unit 44. The microprocessor unit 44 is connected to a garage door operator mechanism 46 which includes the motor which drives the chain 15 to move the door 17 in a conventional manner. The control 38 is connected to the microprocessor 44. A pair of switches 22 and 23 are mounted on a switch unit 19 connected to the unit 11 and also to the microprocessor 44. The switch 22 is a two position switch that can be moved between the operate and program positions to establish the "operate" and "program" modes. The switch 23 can be moved to a number of selected positions indicated by the 1 through 5 so as to allow the particular code of a number of different transmitters to be stored in the receiver so that the receiver will respond to such codes.

In the illustrated embodiment, the receiving unit can respond to up to five different transmitters which have five different transmitting codes. FIG. 5 illustrates the code utilized in which the bit times are nominally 0.5 milliseconds for example. The data times are nominally 1, 2, 3 or 4 milliseconds.

The sync pulse is a unit measure of time. Each datum is measured with respect to the sync pulse and each datum can be 1, 2, 3 or 4 times the length of the sync pulse. The timing is from the rising edge to rising edge of adjacent pulses. Using 10 data bits the number of codes which is available is in excess of one million codes.

In the invention, each transmitter such as transmitters 26 and 28 will have a unique code which is determined by the encoder chip contained in the transmitter. The receiver unit will be able to memorize and store a number of different codes as, for example, five different transmitter codes which eliminates the need of coding switches in either the transmitter or receiver which are used in the prior art. This also eliminates the requirement that the user match the transmitter and receiver code switches.

When the garage door operator is initially installed, the switch 22 is moved to the program mode and the energize button 27 of the first transmitter 26 is depressed so that the unique code of the transmitter 26 is transmitted. This is received by the receiver 41 and decoded by the decoder 43 and supplied to the microprocessor unit 44. The switch 23 is placed in the first position, and with the switch 22 in the program mode the code of the transmitter 26 will be supplied to the memory address storage 47 and stored therein. Then if the switch 22 is moved to the operate mode and the transmitter 26 energized by depressing the transmit switch 27, the receiver 41, decoder and the microprocessor 44 will compare the received code with the code of the transmitter 26 stored in the first memory location in the memory address storage 47 and since the stored memory address for the transmitter 26 coincides with the transmitted code of the transmitter 26 the microprocessor 44 will energize the garage door operation mechanism 46 to open or close the door.

In order to store the code of the second transmitter 28 the switch 22 is moved again to the program mode and the switch 23 to the second position and the transmitter 28 is energized by depressing its transmit switch 29. This causes the receiver 41 and decoder 43 to decode the transmitted signal and supply it to the microprocessor 44 which then supplies the coded signal of the transmitter 28 to the memory address storage 47 where it is stored in a second address storage location. Then the switch 22 is moved to the operate position and when either of the transmitters 26 and 28 are energized, the receiver 41 decoder 43 and microprocessor 44 will energize the garage door operation mechanism 46 to cause the door to either move up or down depending upon its initial position. Thus, the codes of the transmitters 26 and 28 are transmitted and stored in the memory address storage 47 during the program mode after which the garage door operation mechanism will respond to either of the transmitters 26 and 28. Any desired number of transmitters can be programmed to operate the garage door mechanism as, for example, up to five transmitters can be programmed into the memory address storage 47 by using the program switch 22 and the selector switch 23.

This invention eliminates the requirement that binary switches be set in the transmitter or receiver as is done in systems of the prior art to establish a code to which the receiver will respond and the invention also allows a garage door operator to respond to a number of different transmitters because the specific codes of a number of the transmitters is stored and retained in the memory address storage 47 of this unit.

FIGS. 3 and 4 comprise the flow chart which describe both the operate and program modes of the invention. Basically, in the operate mode, a received code must match a program which has already been programmed and for four times so as to operate the garage door. This match is referred to as a valid code in the flow chart. Each valid code can be separated by up to two error codes and still have the output actuate. For example, a code of valid-error-error-valid-valid-valid would actuate the door. On the other hand, a code of valid-valid-valid-error-error-error-valid would not actuate the door.

In the program mode a code must be received four times in a row in order to be permanently stored. Any error code will reset the valid code counter.

with reference to the flow diagrams of FIGS 3 and 4 if it be assumed initially that the switch 22 is in the operate position an incoming signal will be supplied to terminal A in FIG. 3 and an output will be supplied to terminal B which indicates that the switch 22 is not in the program mode but in the operate mode. Terminal B is illustrated in FIG. 4 and the microprocessor compares the incoming code with any codes in the five code locations stored in the memory address storage 47. If these codes match then the error counter is cleared and all other valid counters. If the valid counter receives the code four times than output is supplied to the terminal C which operates the garage door operator. If the valid counter for the code equals less than 4, then the valid code counter is incremented until the valid code counter does not equal 4 which actuates the proper output. Relative to FIG. 4 if the input code does not match any of the five stored codes, then the error counter is incremented and when the error counter equal 3 the error counter is cleared and all valid counters are cleared.

If the switch 22 is in the program mode as shown in FIG. 3 when the incoming signal from a transmitter is received, the flow diagram is followed so as to store the new incoming program in the code location pointed to by the code location pointer 23. It is to be noted that up to five addresses can be stored in the system of the invention.

It is seen that the present invention allows a receiving system to respond to one of a plurality of transmitters which have different unique codes which can be stored in the receiver during a program mode. Each time the "program mode switch" 22 is moved to the program position, a

different storage area as determined by the switch 23 can be connected so that the new transmitter code would be stored in that address After all of the address storage capacity have been used additional codes would erase all old codes in the memory address storage before storing a new one.

Although the invention has been described with respect to preferred embodiments, it is not to be so limited as changes and modifications may be made which are within the full intended scope as defined by the appended claims.

We claim as our invention.

1. A garage door operator for a garage door comprising, a garage door operation mechanism with an output shaft connected to said garage door to open and close it, a radio receiver, a decoder connected to receive the output of said radio receiver, a microprocessor connected to receive the output of said decoder and to said garage door operation mechanism to energize it, a switch moveable between program and operate positions connected to said microprocessor to place said microprocessor in the operate or program mode, a memory means for storing a plurality of addresses connected to said microprocessor when said switch is in the program position, a memory selection switch connected to said microprocessor, a plurality of radio transmitters with different codes, said memory selection switch setable in a first position at a time when a first one of said radio transmitters is energized so that the code of said first transmitter will be stored in said memory means and said memory selection switch set in a second position at a time when a second one of said radio transmitters is energized so that the code of said second transmitter will be stored in said memory means, and said microprocessor placed in the operate mode when said switch is in the operate position so that either or both of said first and second radio transmitters when energized cause said microprocessor to energize said garage door operator mechanism.

2. A garage door operator for a garage door according to claim 1 wherein said first and second radio transmitters when energized radiate coded signals and said microprocessor receives and compares coded signals from said first and second transmitters with coded signals from said first and second transmitters with coded signals stored in said memory means and said microprocessor produces a garage door operate signal if the received transmitted signal and any one of said coded signals stored in said memory means match.

3 A garage door operator according to claim 2 wherein said memory selection switch has "n" positions where "n" is an integer and the codes of "n" transmitters can be stored in said memory means when said switch is in the program mode.

4 A garage door operator according to claim 3 wherein the code stored in said memory means can be changed by placing said switch in the program mode and one of said plurality of transmitters is energized which has a code which differs from the code previously stored in said memory means

5 An operator for controlling operation of equipment comprising a radio receiver, a decoder connected to receive the output of said radio receiver, a microprocessor connected to receive the output of said decoder and to said equipment to energize it, first switch means for selection between program and operate positions connected to said microprocessor to place said microprocessor in the operate or the program mode, a memory means for storing a plurality of addresses connected to said microprocessor when said first switch means is in the program position, a memory selection second switch means connected to said microprocessor, a plurality of radio transmitters with different codes, said memory selection second switch means being adapted to select a first position at a time when a first one of said radio transmitters is energized so that the code of said first transmitter will be stored in said memory means and said memory selection second switch means being adapted to select a second position at a time when a second one of said radio transmitters is energized so that the code of said second transmitter will be stored in said memory means, and said microprocessor placed in the operate mode when said first switch means is in the operate position so that either or both of said first and second radio transmitters, when energized cause said microprocessor to energize said equipment.

6. An operator according to claim 5 wherein said first and second radio transmitters when energized radiate coded signals and said microprocessor receives and compares coded signals from said first and second transmitters with coded signals stored in said memory means and said microprocessor produces an operate signal if the received transmitted signal and any one of said coded signals stored in said memory means match.

7 An operator according to claim 6 wherein said second switch means, has "n" positions where "n" is an integer and the codes of "n" transmitters can be stored in said memory means when said first switch means is in the program mode.

8. An operator according to claim 7 wherein the code stored in said memory means can be changed by placing said first switch means in the program mode and one of said plurality of transmitters is energized which has a code which differs from the code previously stored in said memory means.

* * * * *

of the Brewster machine pointed out numerous features which were distinguishable from the preferred embodiment of the invention. However, only one of these features appears as an express limitation of each of the disputed claims, namely, intermixing the microingredients in the liquid before discharging them into the feed ration. The Brewster system did not intermix the microingredients in the liquid carrier—presumably because such intermixing would interfere with the weighing operation.

This court notes that the prior art did not "perform[ ] the claimed method" as the district court suggests. Rather the "claimed method" is the unique combination of steps found in method claims 63, 93, and 94—not any single step in isolation. The prosecution history suggests that the applicant avoided the prior art by including an intermixing limitation in the claims. The prosecution history does not suggest that the applicant disavowed the weigh dump method. The district court erred by interpreting these statements in the background section of the patent and the prosecution history as a disavowal of the weigh dump method.

Mr. Hummel's Personal Liability

[15–17] Officers of an allegedly infringing corporation can be held personally liable for actively inducing infringement under 35 U.S.C. § 271(b) only if they "knew or should have known [their] actions would induce actual infringements." *Manville Sales Corp. v. Paramount Sys., Inc.*, 917 F.2d 544, 553, 16 U.S.P.Q.2d 1587, 1593 (Fed.Cir.1990). Pre-issuance activities alone cannot establish inducement to infringe. *See National Presto Indus., Inc. v. West Bend Co.*, 76 F.3d 1185, 1196, 37 U.S.P.Q.2d 1685, 1693 (Fed.Cir.1996). Micro Chemical brought this suit two days after the patent issued, at which time Mr. Hummel took reasonable steps to avoid infringement. Among other things, Mr. Hummel sought and relied on the advice of counsel in redesigning the accused machines. The undisputed facts do not establish the knowledge necessary to find inducement to infringe. The district court appropriately granted summary judgment finding Mr. Hummel not personally liable for inducing Lextron's infringement.

III.

In conclusion, this court reverses the district court's entry of judgment of non-infringement with respect to claims 63, 74, 93, and 94, but affirms with respect to claim 91. This court affirms the district court's grant of summary judgment finding Mr. Hummel not personally liable for inducement to infringe. The case is remanded to the district court for an assessment of damages.

COSTS

Each party shall bear its own costs.

AFFIRMED-IN-PART, REVERSED-IN-PART, and REMANDED.

OVERHEAD DOOR CORPORATION and GMI Holdings, Inc., Plaintiffs–Appellees,

v.

The CHAMBERLAIN GROUP, INC., Defendant–Appellant.

No. 98–1428.

United States Court of Appeals, Federal Circuit.

Oct. 13, 1999.

Rehearing and Suggestion for Rehearing En Banc Denied Dec. 16, 1999.

Manufacturer of remote control garage door opener systems brought declaratory judgment action asserting that manufacturer's "Intellicode" system did not

infringe patents. Patentee counterclaimed, alleging willful infringement. The United States District Court for the Northern District of Texas, Sidney A. Fitzwater, J., granted summary judgment of noninfringement, and patentee appealed. The Court of Appeals, Rader, Circuit Judge, held that: (1) term "memory selection switch" meant a mechanical device, separate from the microprocessor, with different positions; (2) first claim at issue was not literally infringed, but fact issues existed as to whether it was infringed under doctrine of equivalents; (3) term "memory selection switch means" encompassed both a mechanical switch and a software-based embodiment; (4) fact issues existed as to whether second disputed claim was literally infringed; and (5) "switch moveable" element, "first switch means" element, and "different codes" element were present in accused system.

Affirmed in part, vacated in part, and remanded.

1. Federal Courts ⇐766

Court of Appeals reviews a district court's grant of summary judgment without deference.

2. Patents ⇐324.5

Court of Appeals reviews the district court's construction of a patent claim without deference.

3. Patents ⇐314(5)

Whether accused device exactly contains each element of a patent claim, as properly construed, or its equivalent, is a determination of fact.

4. Patents ⇐101(2)

Term "memory selection switch," in patent for remote control garage door opener system, meant a mechanical device, separate from the microprocessor, with different positions, each position corresponding to a different location in memory, thus enabling the garage door operator to store codes in different memory locations.

5. Patents ⇐226.6

Literal infringement of a patent claim requires that every limitation recited in the claim appear in the accused device, that is, that the properly construed claim reads on the accused device exactly.

6. Patents ⇐238

Patent for remote control garage door opener system containing element of mechanical "memory selection switch" was not literally infringed by accused system that selected memory locations with a software program, not a mechanical switch.

7. Patents ⇐237

Finding of infringement under "doctrine of equivalents" requires that the accused product contain each limitation of the patent claim or its equivalent.

See publication Words and Phrases for other judicial constructions and definitions.

8. Patents ⇐237

An element in the accused product is equivalent to a patent claim element, for purpose of doctrine of equivalents, if the differences between the two are insubstantial to one of ordinary skill in the art.

9. Patents ⇐323.2(2)

Although equivalence is a factual matter normally reserved for a factfinder, on claim of patent infringement under doctrine of equivalents, the trial court should grant summary judgment in any case where no reasonable factfinder could find equivalence.

10. Patents ⇐323.2(3)

Genuine issue of material fact as to whether one of ordinary skill in the art would find accused remote control garage door opener system's software-driven memory selection system insubstantially different from hardware switch claimed in patent for such system precluded summary judgment for manufacturer of accused system, on claim that accused system infringed patent under doctrine of equivalents.

11. Patents ⇐226.7

The "function-way-result test" dictates that an element in the accused device is equivalent to a patent claim element if it performs substantially the same function in substantially the same way to obtain the same result.

See publication Words and Phrases for other judicial constructions and definitions.

12. Patents ⇐226.7

District court's finding that function of memory selection switch in claimed remote control garage door opener system was to permit selection of particular memory location at the receiver for subsequent storage of transmitted code by use of a switch connected to the microprocessor, in applying function-way-result test to claim of infringement under doctrine of equivalents, improperly incorporated claim element's "way" into definition of function, effectively limiting the claim element to its literal terms.

13. Patents ⇐101(2)

In means-plus-function element of patent claiming remote control garage door opener system, claimed "memory selection switch means" encompassed both a mechanical switch and a software-based embodiment, in view of the patent's written description and its prosecution history.

14. Patents ⇐226.7

Where patent claim element utilizes term "means" and claim does not specify any structure or material for performing the recited function, that element is a means-plus-function element under patent statute, and that means covers the corresponding structure, material, or acts described in the specification and equivalents thereof. 35 U.S.C.A. § 112.

15. Patents ⇐324.5

District court's determination of corresponding structure, in connection with means-plus-function patent claim, is a matter of claim construction, which Court of Appeals reviews de novo. 35 U.S.C.A. § 112.

16. Patents ⇐323.2(3)

Genuine issue of material fact as to whether accused remote control garage door opener system used structure equivalent to "memory selection switch means" encompassed by patent precluded summary judgment for patentee on its claim that accused system literally infringed the patent.

17. Patents ⇐226.7

An accused device satisfies a means-plus-function element literally if it performs the identical function required by the limitation, and incorporates the structure disclosed in the specification or an equivalent thereof.

18. Patents ⇐226.7

A structure in an accused device is equivalent to the disclosed structure corresponding to a means-plus-function element in a patent claim if it is insubstantially different from the disclosed structure.

19. Patents ⇐101(2)

Term "switch moveable" in patent for remote control garage door opener system, which represented program/operate switch, meant a moveable switch connected to the microprocessor having at least two positions, and did not require that each switch position be stationary and completely user-selected; thus, that element was present in accused system, which had two-position, spring-loaded, push-button switch, even though accused switch returned to stationary position when user released the push-button.

20. Patents ⇐101(2)

"First switch means" for selecting between program and operate positions, in patent claiming remote control garage door opener system, covered a two-position, mechanical switch or its structural equivalent, and was present in accused system, which had two-position, spring-loaded, push-button switch.

21. Patents ⇐101(2)

Term "different codes" in patent for remote control garage door opener system meant factory-defined codes stored within the radio transmitters which uniquely identified each different transmitter and were not selectable or modifiable by user of system, and term did not require an identical code in the transmitter and the receiver; thus, that element was present in accused system in which unique, fixed, factory-programmed string of binary numbers in transmitter did not exactly match bits stored in microprocessor's memory, but its microprocessor, in "program" mode, stored secret key and thereafter, in "operate" mode, used secret key to verify authorized transmitters.

Kenneth R. Glaser, Akin, Gump, Strauss, Hauer & Field, L.L.P., Dallas, Texas, argued for plaintiffs-appellees. With him on the brief were Michael Lowenberg, Steven E. Ross, and Alex Chartove.

John F. Flannery, Fitch, Even, Tabin & Flannery, Chicago, Illinois, argued for defendant-appellant. With him on the brief were Robert J. Fox and Karl R. Fink.

Before MICHEL, RADER, and SCHALL, Circuit Judges.

RADER, Circuit Judge.

The United States District Court for the Northern District of Texas ruled on summary judgment that Overhead Door Corp. and GMI Holdings, Inc. did not infringe U.S. Patent No. Re. 35,364 (the '364 patent). *See Overhead Door Corp. v. Chamberlain Group, Inc.*, No. 95–CV–1648–D (N.D.Tex. Apr. 30, 1998) (order). The '364 patent claims improvements on remote control systems for garage door openers. Because the district court erred in concluding as a matter of law that the claims at issue cannot cover the accused device under the doctrine of equivalents or as structural equivalents, this court affirms-in-part, vacates-in-part, and remands.

I.

Chamberlain Group, Inc. owns the '364 patent. Both Chamberlain and Overhead Door manufacture and sell remote control garage door opener systems. Remote control garage door opener systems typically include hand-held, portable transmitters and a stationary garage door opening motor with a processing unit and receiver. To open or close a garage door, a user presses a button on the transmitter to send a signal to the receiver. The receiver relays the signal to a processing unit that directs the door motor to open or close the garage door.

To prevent signals of foreign transmitters from opening the door, these systems use coded signals. A unique code thus links each transmitter to its own system, and a garage door opener's processing unit verifies that the signal code comes from its own transmitter before activating the opening motor.

Before the '364 patent, garage door systems required users to install transmitter codes manually. A typical code installation system required the user to set matching dual in-line package (DIP) switches on the transmitters. These manual codes had shortcomings, including installation errors by inexperienced installers and limits on code length due to the size of DIP switches.

The '364 patent eliminates manual code switches in garage door transmitters, and enables a garage door opener to learn the identities of and respond to multiple transmitters with different codes. *See* '364 patent, col. 4, ll. 15–22. An embodiment of the invention described in the '364 patent includes two or more transmitters with lengthy, factory-programmed codes. The microprocessor in the receiver switches between "program" and "operate" modes. In the "program" mode, the microprocessor stores in a memory a pre-programmed code it receives from the transmitter. *See id.* at col. 3, ll. 40–48. The microprocessor

can store multiple codes in this manner. In the "operate" mode, the microprocessor verifies that a signal matches one of the stored transmitter codes. *See id.* at col. 3, ll. 49–58.

The '364 patent includes eight claims, all of which Chamberlain asserts in this action. Claims 1–4 were part of the original patent, U.S. Patent No. 4,750,118 (the '118 patent). Chamberlain added claims 5–8 during reissue proceedings. The two independent claims 1 and 5 recite:

1. A garage door operator for a garage door comprising,

a garage door operation mechanism with an output shaft connected to said garage door to open and close it,

a radio receiver,

a decoder connected to receive the output of said radio receiver,

a microprocessor connected to receive the output of said decoder and to said garage door operation mechanism to energize it,

a switch moveable between program and operate positions connected to said microprocessor to place said microprocessor in the operate or program mode,

a memory means for storing a plurality of addresses connected to said microprocessor when said switch is in the program position,

a memory selection switch connected to said microprocessor,

a plurality of radio transmitters with *different codes, said memory selection switch setable in a first position* at a time when a first one of said radio transmitters is energized so that the code of said first transmitter will be stored in said memory means and *said memory selection switch set in a second position* at a time when a second one of said radio transmitters is energized so that the code of said second transmitter will be stored in said memory means,

and said microprocessor placed in the operate mode when said switch is in the operate position so that either or both of said first and second radio transmitters when energized cause said microprocessor to energize said garage door operator mechanism.

5. An operator for controlling operation of equipment comprising:

a radio receiver,

a decoder connected to receive the output of said radio receiver,

a microprocessor connected to receive the output of said decoder and to said equipment to energize it,

first switch means for selection between program and operate positions connected to said microprocessor to place said microprocessor in the operate or the program mode,

a memory means for storing a plurality of addresses connected to said microprocessor when said first switch means is in the program position,

a memory selection second switch means connected to said microprocessor,

a plurality of radio transmitters with *different codes, said memory selection second switch means being adapted to select a first position* at a time when a first one of said radio transmitters is energized so that the code of said first transmitter will be stored in said memory means and *said memory selection second switch means being adapted to select a second position* at a time when a second one of said radio transmitters is energized so that the code of said second transmitter will be stored in said memory means,

and said microprocessor placed in the operate mode when said first switch means is in the operate position so that either or both of said first and second radio transmitters, when energized cause said microprocessor to energize said equipment.

'364 patent, col. 5, ll. 11–35; col. 6, ll. 7–30 (emphasis added).

Overhead Door's accused openers—marketed as the "Intellicode" system—also use factory-programmed identification codes instead of manual switches and "learn" to identify multiple transmitters. Like the invention claimed in the '364 patent, the Intellicode features "program" and "operate" modes, and stores transmitter codes in selected memory locations during the learning process. The Intellicode, however, does not use a manual, mechanical memory selection switch. Rather, the Intellicode features software that determines the memory location for each new code. The Intellicode's microprocessor, under control of the software, identifies unused locations in memory and automatically stores a new code in an unused and available location. The parties dispute whether the Intellicode's software-driven memory selection scheme is outside the scope of the '364 patent.

On August 7, 1995, Overhead Door filed a declaratory judgment action, seeking a judgment that the Intellicode does not infringe the '118 patent. Chamberlain counterclaimed, alleging willful infringement. The '118 patent reissued as the '364 patent in October 1996 and Overhead Door amended its complaint to include additional claims 5–8. The parties filed cross-motions for summary judgment. The court referred the motions to a special master for a recommendation.

The special master concluded as a matter of law that the Intellicode does not literally infringe the '364 patent. In reaching this conclusion, the special master construed the claim elements "memory selection switch" (claim 1) and "memory selection second switch means" (claim 5) to require "a switch *separate from* the microprocessor which is *user operated* to select different positions of the switch." *Overhead Door*, No. 95–CV–1648–D, at 36 (N.D.Tex. Jan. 13, 1999) (report of Special Master) (emphasis added). The special master further determined that the claims "[can]not be construed so broad[ly] to include a microprocessor controlled random memory selection process." Because the Intellicode selects memory locations with software rather than a manual switch, the special master concluded that the Intellicode does not literally infringe the claims in suit.

The special master also concluded as a matter of law that the Intellicode does not infringe the '364 patent under the doctrine of equivalents. The special master identified the "function" of the "memory selection switch" and "memory selection second switch means" as "permit[ting] selection of a particular memory location at the receiver for subsequent storage of a transmitted code." *Id.* at 46. According to the special master, the Intellicode lacks this function because its microprocessor uses software to automatically select memory locations. Further, the special master noted that "the user is unable to predetermine memory location[s]" or replace a location with a new code. The special master therefore concluded that the Intellicode cannot infringe the '364 patent under the doctrine of equivalents.

The district court adopted the special master's report in its entirety, thereby granting Overhead Door's summary judgment motion of non-infringement and denying Chamberlain's cross-motion. Chamberlain appeals.

II.

[1–3] This court reviews a district court's grant of summary judgment without deference. *See Chiuminatta Concrete Concepts, Inc. v. Cardinal Indus., Inc.*, 145 F.3d 1303, 1307, 46 USPQ2d 1752, 1755 (Fed.Cir.1998). This court also reviews the district court's claim construction without deference. *See Cybor Corp. v. FAS Technologies, Inc.*, 138 F.3d 1448, 1456, 46 USPQ2d 1169, 1172 (Fed.Cir.1998) (*en banc*). Whether the accused device exactly contains each element, as properly construed, or its equivalent, is a determination of fact. *See Southwall Technologies, Inc. v. Cardinal IG Co.*, 54 F.3d 1570, 1575, 34 USPQ2d 1673, 1676 (Fed.Cir.1995). In re-

view of the district court's summary judgment in favor of Overhead Door, this court draws reasonable inferences from the evidence in favor of the non-movant, Chamberlain. *See Anderson v. Liberty Lobby, Inc.*, 477 U.S. 242, 255, 106 S.Ct. 2505, 2513, 91 L.Ed.2d 202 (1986).

In dispute are three elements of claim 1: "memory selection switch," "switch moveable," and "different codes." Similar elements of claim 5 are in dispute. The first two elements of claim 5, however, are in means-plus-function format, i.e., "memory selection second switch means" and "first switch means."

"Memory Selection Switch" – Claim 1

[1] The district court construed the "memory selection switch" element of claim 1 to mean "a switch separate from the microprocessor which is user operated to select different positions of the switch." The claim uses the term "switch," a word connoting a mechanical device with different settings, such as "on" or "off." Claim 1 further defines the memory selection switch as "connected to" the microprocessor, "setable in" a first position, and "set in" a second position. This claim language is more consistent with a mechanical switch attached to the microprocessor, rather than software programmed into the microprocessor.

In the '364 patent's "Brief Description of the Drawings," the patentee states that "Fig. 2 illustrates in block form *the invention*." Col. 2, l. 43 (emphasis added). Figure 2 complements the definition supplied by the claim language:

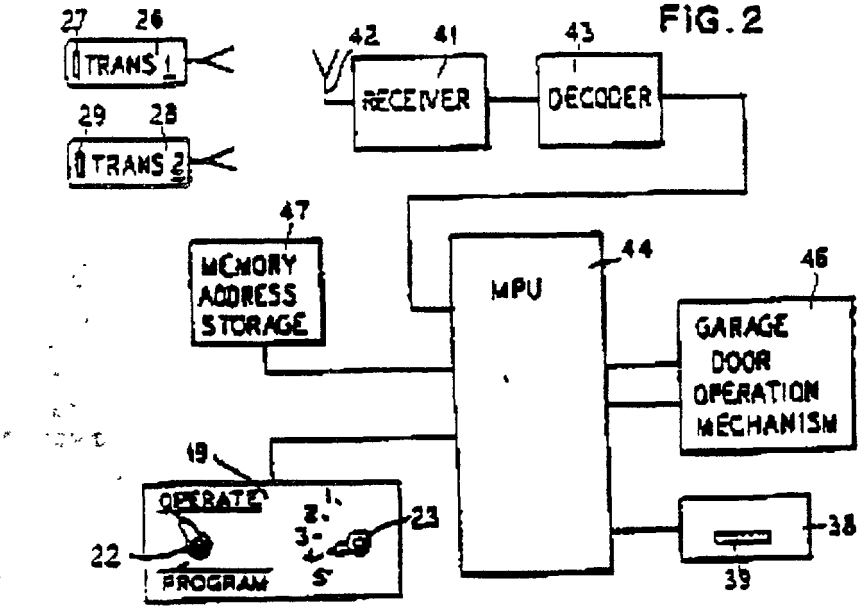


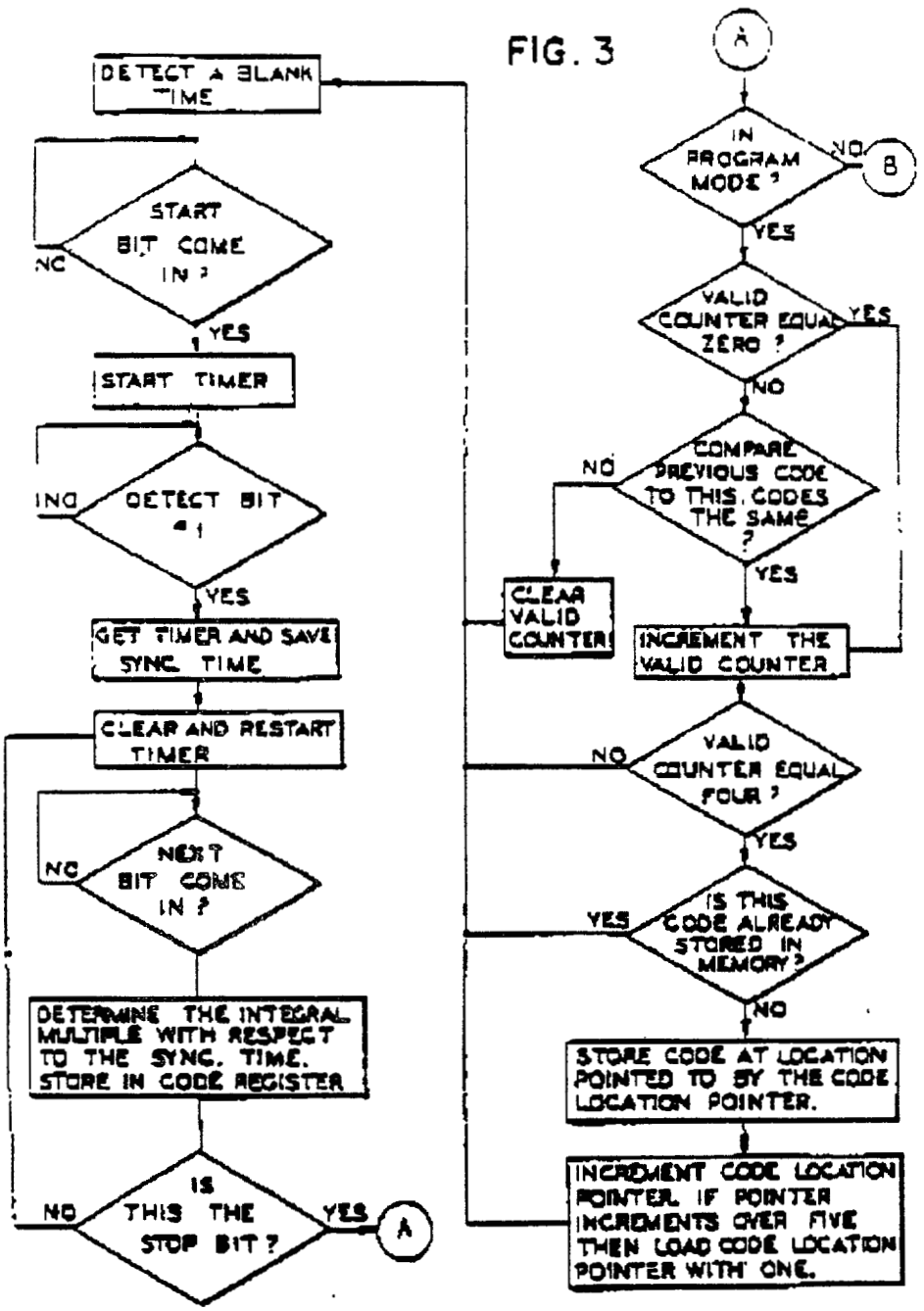
Figure 2 shows the program/operate switch 22 as a mechanical switch, which alternates between "program" and "operate." Moreover, the drawing depicts the memory selection switch 23 as a mechanical switch with five separate numbered positions. The associated written description identifies the memory selection switch with 23, *see* '364 patent, col. 3, ll. 9–11, and describes it as a five-position, moveable switch separate from and "*connected to*" the microprocessor 44. *See id.* at col. 3, ll. 9–19 (emphasis added); *see also* Fig. 2. Again this part of the patent suggests a mechanical switch.

Finally, Chamberlain does not contest the special master's interpretation of the program mode "switch" as a mechanical toggle switch. To interpret the term "switch" consistently in the claim and to harmonize the drawing depiction with the

claim language, this court confirms the district court's reading of the term "switch." Thus, the term "memory selection switch" means a mechanical device separate from the microprocessor. This interpretation is also most in harmony with the prosecution history of the reissue application, as explained later.

In reaching this claim interpretation, this court considered but rejected the contention that Figure 3 discloses as part of the claim a software embodiment for the switch of claim 1. Figure 3 of the '364 patent, shown below, illustrates how the invention receives and validates codes:

FIG. 3



In Figure 3, the two dialog boxes in the lower-right corner refer to storing the code at the location "pointed to" by the "code location pointer," "increment[ing]" the code location pointer, and "load[ing]"

the code location pointer with a value of one. The '364 patent, however, does not indicate whether the "code location pointer" is a particular embodiment of the "switch" of claim 1, or some other (unclaimed) component. "Code location pointer" appears nowhere in the claims. Moreover, the only reference to a code location pointer in the written description is a single sentence that does not illuminate Figure 3: "If the switch 22 is in the 'program' mode as shown in FIG. 3 when the incoming signal from a transmitter is received, the flow diagram is followed so as to store the new incoming program in the code location pointed to by the code location pointer 23." Col. 4, ll. 57–61. The vague terms in Figure 3 do not override the claim language and written description that closely identify the "memory selection switch" as a mechanical device. This court interprets "memory selection switch" to mean a mechanical switch with different positions, each position corresponding to a different location in memory, thus enabling the garage door operator to store codes in different memory locations. Thus, this court affirms on review the district court's interpretation of "memory selection switch."

[5, 6] Applying this claim construction to the accused device, this court affirms the district court's summary judgment of no literal infringement of claim 1. "Literal infringement of a claim requires that every limitation recited in the claim appear in the accused device, i.e., that the properly construed claim reads on the accused device exactly." *Amhil Enters., Ltd. v. Wawa, Inc.*, 81 F.3d 1554, 1562, 38 USPQ2d 1471, 1476 (Fed.Cir.1996). Claim 1 covers a mechanical "memory selection switch." The accused Intellicode system, in contrast, selects memory locations with a software program, not with a mechanical switch. Thus, the "memory selection switch," as correctly construed, is literally absent from the Intellicode. Therefore, the Intellicode does not literally infringe claim 1.

[7, 8] The district court erred, however, in also deciding on summary judgment that the Intellicode did not infringe under the doctrine of equivalents. The doctrine of equivalents requires that the accused product contain each limitation of the claim or its equivalent. *See Warner–Jenkinson Co. v. Hilton Davis Chem. Co.*, 520 U.S. 17, 40, 117 S.Ct. 1040, 137 L.Ed.2d 146, 41 USPQ2d 1865, 1875 (1997). An element in the accused product is equivalent to a claim element if the differences between the two are "insubstantial" to one of ordinary skill in the art. *Warner–Jenkinson*, 520 U.S. at 39–40, 117 S.Ct. 1040; *Hilton Davis Chem. Co. v. Warner–Jenkinson Co.*, 62 F.3d 1512, 1517, 35 USPQ2d 1641,-1644 (Fed.Cir.1995) (*en banc*), *rev'd on other grounds*, 520 U.S. 17, 117 S.Ct. 1040, 137 L.Ed.2d 146 (1997). The district court found as a matter of law that the Intellicode had no equivalent of the memory selection switch claim element.

[9, 10] "Although equivalence is a factual matter normally reserved for a fact-finder, the trial court should grant summary judgment in any case where no reasonable fact-finder could find equivalence." *Sage Prods., Inc. v. Devon Indus., Inc.*, 126 F.3d 1420, 1423, 44 USPQ2d 1103, 1106 (Fed.Cir.1997). This case does not satisfy this lofty standard. The record contains considerable evidence, including several reports and declarations by Chamberlain's expert, Dr. Rhyne, that one of ordinary skill in the art would find the Intellicode's software-driven memory selection system insubstantially different from the hardware switch of claim 1. Dr. Rhyne averred in his June 2, 1997 report: "[it is a] fundamental and well understood tenet of the computing art [that] . . . '[a]ny software process can be transformed into an equivalent hardware process, and any hardware process can be transformed into an equivalent software process.'" *See* ED KLINGLER, MICROPROCESSOR SYSTEMS DESIGN 5 (1977). Dr. Rhyne stated that this "dualistic transformation," known as the "hardware/software" tradeoff, effectively

means that the selection of a software pointer for a microprocessor versus a hardware switch to control a microprocessor-based system is simply a matter of design choice. This record evidence shows that one of skill in the art would recognize these alternative systems as interchangeable substitutes. Drawing all reasonable inferences in favor of Chamberlain, as this court must in reviewing the summary judgment of non-infringement, this court concludes that Dr. Rhyne's statements and supporting citations to computer science literature show a genuine issue of material fact precluding summary judgment.

In discerning this genuine factual issue, this court also considered the district court's interpretation that a mechanical switch would necessarily require a human operator. In operation of a mechanical switch, a human operator would indeed set the memory selection switch to one of five positions. This "user operated" characteristic of a mechanical switch, however, would not necessarily preclude a finding that software performs equivalently without human operation. Indeed in other contexts, this court has noted the interchangeability of hardware and software. *See, e.g., Pennwalt Corp. v. Durand–Wayland, Inc.*, 833 F.2d 931, 935, 4 USPQ2d 1737, 1740 (Fed.Cir.1987) (*en banc*) ("If . . the accused devices differ only in substituting a computer for hard-wired circuitry, [the patentee] might have a stronger position for arguing that the accused devices infringe the claims."). Moreover the Supreme Court has acknowledged that interchangeability can be one of the hallmarks of an equivalent. *See Warner–Jenkinson*, 520 U.S. at 37, 117 S.Ct. 1040 ("known interchangeability . . . for an element of a patent is one of the express objective factors . . . bearing upon whether the accused device is substantially the same as the patented invention"); *Graver Tank & Mfg. Co., Inc. v. Linde Air Products Co.*, 339 U.S. 605, 609, 70 S.Ct. 854, 94 L.Ed. 1097, 85 USPQ 328, 331 (1950) ("An important factor [in determining equivalency] is whether persons reasonably skilled in the art would have known of the interchangeability ").

[11, 12] This court has explained that the "function-way-result" test may help detect an equivalent, particularly for mechanical elements. *See Dawn Equipment Co. v. Kentucky Farms Inc.*, 140 F.3d 1009, 1016, 46 USPQ2d 1109, 1113 (Fed.Cir. 1998). The function-way-result test dictates that an element in the accused device is equivalent to the claim element if it "performs substantially the same function in substantially the same way to obtain the same result." *Graver Tank*, 339 U.S. at 608, 70 S.Ct. 854. Applying the function-way-result test to claim 1, the district court found that "the function of the memory selection switch [is] . . to permit selection of a particular memory location at the receiver for subsequent storage of a transmitted code, *by use of a switch connected to the microprocessor.*" *Overhead Door Corp*, No. 95–CV–1648–D, at 24 (N.D.Tex. Jan. 13, 1999) (emphasis added). The district court then found that this function was "totally missing" from the accused device because "the user is unable to predetermine selection of specific, desired memory locations." This application of the function-way-result test erroneously incorporates the claim element's "way" into the definition of the function, effectively limiting the claim element to its literal terms.

The claim language and the specification explain that the "memory selection switch" functions to select memory locations. This claim element accomplishes its function by way of a mechanical switch. This particular switch constitutes the claim element's "way" of accomplishing the memory selection function, not the function itself. The result of this element is storage of codes in different memory locations. The record at this stage, preliminary to a trial, creates a genuine issue of material fact whether the Intellicode accomplished substantially the same function, in substantially the same

way, to achieve substantially the same result.

Moreover, contrary to the district court's determination, this court's ruling in *Sage Products* does not limit the range of equivalents to the memory selection switch in this case. In *Sage Products*, this court noted that the claim limitations "top of the container" and "over said slot" constituted "a precise arrangement of structural elements" and a "clear structural limitation" in a "relatively simple structural device." *Sage Prods.*, 126 F.3d at 1425. Finding that Sage's theory of equivalence – i.e., a container having two constrictions below its top was equivalent to the claimed container having a constriction above and a constriction below—would "remove entirely the 'top of the container' and 'over said slot' limitations from the claim," this court affirmed the district court's grant of summary judgment of non-infringement. *Id.* at 1423–24. Thus, the proposed application of the doctrine in *Sage Products* would have utterly written out of the claim not one, but at least two (maybe more) express limitations of the claim. Indeed under Sage Products' equivalents theory, a finding of equivalents for one limitation ("at the top") would necessarily require writing out of the claim another limitation ("over said slot"). No matter how the patentee purported to apply the claim to the accused device under the doctrine, the device was always missing at least one limitation. Thus, the claim language specifically negated the patentee's equivalence theory. Moreover, this court in *Sage Products* noted that "any subsequent change in the state of the art, such as later developed technology" would have been eligible for coverage under the doctrine of equivalents, thus clearly defining at least one type of expanded claim coverage under the doctrine. *Id.* at 1425.

In contrast to the facts in *Sage Products*, claim 1 of the '364 patent does not contain any clear structural limitations that preclude a reasonable jury from finding a software system equivalent to the claimed system. By definition, an equivalent does not fall literally within the claim language. Although the literal meaning of the "memory selection switch" does not cover the software-implemented Intellicode, this case does not preclude application of the doctrine under *Sage Products* because any application of the doctrine would not leave some aspect of the claim missing from the accused device. Applying the doctrine of equivalents to cover Intellicode's software does not vitiate the "memory selection switch" element. *See id.*, at 1423–24.

As properly construed, the "memory selection switch" means a mechanical switch for selecting memory locations. The question remains whether the Intellicode's software-driven memory selection scheme is equivalent to a mechanical switch. This issue remains for the fact-finder to determine at trial in view of the considerable evidence in the record.

"Memory Selection Second Switch Means" – Claim 5

[13, 14] Claim 5 recites a "memory selection second switch means being adapted to select a first position .. and a second position." Because this claim element utilizes the term "means" and the claim does not specify any structure or material for performing the recited function, the district court properly held "memory selection second switch means" is a means-plus-function element under 35 U.S.C. § 112, ¶ 6 (1994). *See Al-Site Corp. v. VSI Int'l, Inc.*, 174 F.3d 1308, 1318, 50 USPQ2d 1161, 1166 (Fed.Cir. 1999) ("if the word 'means' appears in a claim element in combination with a function, it is presumed to be a means-plus-function element"). Thus, "memory selection second switch means" covers the "corresponding structure, material, or acts described in the specification and equivalents thereof." 35 U.S.C. § 112, ¶ 6 (1994).

[15] The district court's determination of corresponding structure is a matter of claim construction, *see Chiuminatta*, 145

F.3d at 1306, which this court reviews *de novo*. *See Cybor*, 138 F.3d at 1456. Determining whether Figure 3 is a "corresponding structure" for the "switch means" of claim 5 requires the court to consult again the language of the claim and the other factors that inform claim meaning. Of course, the central focus remains on the claim language. The written description, the prosecution history, and admissible extrinsic evidence may supply context to understand the claim language. *See Vitronics Corp. v. Conceptronic, Inc.*, 90 F.3d 1576, 1582, 39 USPQ2d 1573, 1576–77 (Fed.Cir.1996); *Pitney Bowes, Inc. v. Hewlett–Packard Co.*, 182 F.3d 1298, 1309, 51 USPQ2d 1161, 1169 (Fed. Cir.1999).

In construing claim 5, the district court determined the term "memory selection second switch means" encompasses the same scope as the "memory selection switch" of claim 1. Specifically, the district court determined that "memory selection second switch means" covers only the mechanical switch of Figure 2, not the software embodiment of Figure 3. The written description of the '364 patent and the prosecution history, however, reveal a broader meaning of "memory selection switch means."

As previously explained in this court's analysis of claim 1, Figure 3 illustrates a flow diagram "describ[ing] both the operate and program modes of the invention." '364 patent, col. 4, ll. 23–24. The two lower-right corner dialog boxes of Figure 3 describe steps to "store code at location *pointed to* by the code location *pointer*" and "*increment* code location pointer[;] if pointer increments over five then *load* code location pointer with one." *See id.* at Fig. 3 (emphasis added). Dr. Rhyne's expert testimony shows that one of ordinary skill in the computer science art would understand the underlined terms to describe software operations.

Although software operations do not fall within the literal scope of the "memory selection switch" in claim 1, the reissue prosecution history also discloses a broader reading for the "switch means" of claim 5. First, the patentees' representation to the Patent and Trademark Office in its November 29, 1989 sworn declaration indicated their intent to include the algorithm of Figure 3 as a "corresponding structure" for the switch means. The patentees stated:

> We believe the aforesaid Letters Patent to be wholly or partly inoperative or invalid by reason of our *claiming less than we had a right to claim* in the patent. More specifically, we believe the sole independent original claim [i.e., claim 1 of the '118 and now claim 1 of the '364 patent] is *too narrow* in three respects:
>
> . . . .
>
> (c) The claim requires a "switch moveable" and a "*memory selection switch*" but should have required a—first switch means—and a—*memory selection switch means*—, respectively, because switch means *includes electronic switches as well as mechanical switches.*

J.A. at 5388 (emphasis added). While this statement weighs against construing claim 1 to include software operations, it gives a broader reading to claim 5. This statement evinces the patentees' use of the term "switch means" to include microprocessor operations driven by software, i.e., "electronic" switches, as opposed to a mechanical switch of Figure 2. The patentees' use in claim 5 of the term "switch means" rather than "switch" and "being adapted to select" rather than "setable" and "set," to describe software operations, further support a broader construction.

Later in the reissue proceedings, the patentees argued in response to an anticipation rejection:

> Applicants' method and apparatus is intended to simplify the remote control of equipment by code transmitters . Such simplifications are provided by including multiple storage locations in the receiver and including a *programming*

routine which receives and stores codes transmitted from the code transmitters of the system.

J.A. at 5822 (emphasis added). This statement further supports reading "switch means" to include structure corresponding to Figure 3.

The differences in claim language, bolstered by the patentees' statements during the reissue proceedings, cause this court to reach a broader construction for claim 5 than for claim 1. *See Vitronics*, 90 F.3d at 1582. The district court erred in ruling that only the mechanical switch in Figure 2 is "corresponding structure" for the claimed "switch means." "Switch means," when properly construed, also covers the software-based embodiment described in Figure 3.

[16, 17] Chamberlain asserts that, should this court construe claim 5 to cover the software embodiment of Figure 3, then it is entitled to summary judgment of literal infringement. An accused device satisfies a means-plus-function element literally if it performs the identical function required by the limitation, and incorporates the structure disclosed in the specification or an equivalent thereof. *See Cybor*, 138 F.3d at 1456. The language of claim 5 and the written description establish that the function of "switch means" is to select different memory locations, thereby enabling the microprocessor to store transmitter identifiers in the memory locations. The parties do not dispute that the Intellicode's memory selection software program performs this function.

The Intellicode, however, constitutes a different "structure" than the software disclosed in the '364 patent because it uses a different algorithm to perform the recited function. Figure 3 and the corresponding description indicate that the code location pointer increments through a series of memory locations, automatically erasing the previous contents of a memory location when it stores a new transmitter code in that location. When the pointer can no longer increment, i.e., when it is pointing to the last memory location in the series, the microprocessor "load[s] code location pointer with one," causing the pointer to loop back and select the first memory location in the series. The Intellicode, on the other hand, randomly chooses an unused memory location. Thus, the Intellicode's memory selection scheme is not identical to the structure disclosed in Figure 3 of the '364 patent.

[18] Moreover, Overhead Door presented evidence that its memory selection scheme is not structurally equivalent to that of Figure 3. A structure in an accused device is equivalent to the disclosed structure corresponding to a means-plus-function element if it is insubstantially different from the disclosed structure. *See Chiuminatta*, 145 F.3d at 1309. Overhead Door urged that the Intellicode software is substantially different from the claim element because its software uses memory more efficiently and minimizes the chances of overwriting previously-stored codes. Viewing the evidence in a light most favorable to the non-movant Overhead Door, this court finds that Overhead Door has raised a genuine issue of material fact precluding summary judgment of literal infringement of claim 5. Accordingly, this court remands for the fact-finder to determine whether the Intellicode uses a structure equivalent to the mechanical switch in Figure 2 or to the software-implemented algorithm in Figure 3, for selecting different memory locations.

"Switch Moveable" – Claim 1

[19] This court also agrees with the district court that "switch moveable" of claim 1, i.e., the program/operate switch, is present in the accused device. The district court properly construed this element as a "moveable[ ] switch connected to the microprocessor having at least two positions." The mechanical switch of Figure 2 supports this construction. Contrary to Overhead Door's assertion, neither the claim language nor the specification re-

quires each switch position to be stationary and completely user-selected.

Applying the trial court's correct claim construction to the accused device, this court affirms the district court's finding that the Intellicode's two-position, spring-loaded, push-button switch satisfies the "switch moveable" limitation. In affirming, this court understands that the accused switch returns to a stationary position when the user releases the push-button. The parties do not dispute that the program/operate switch of the Intellicode is "a mechanical, two position, user-operated switch." Overhead Door urges, however, that its momentary switch does not have an operate position, the program position is not stationary, and the switch does not permit the user to select the "operate" mode. This court finds this argument unpersuasive. The Intellicode two-position switch is "a mechanical switch movable between an operate and program position at least momentarily and is connected to the microprocessor." Thus, this court affirms the district court's summary judgment ruling insofar as it rested on the finding that the Intellicode literally meets the "switch moveable" requirement.

"First Switch Means" – Claim 5

[20] Claim 5 recites "first switch means for selection between program and operate positions connected to said microprocessor." This claim element uses the term "means" and the claim does not specify any structure or material for performing the recited function. Therefore, the district court properly held "first switch means" is a means-plus-function element under 35 U.S.C. § 112, ¶ 6. *See Al-Site*, 174 F.3d at 1318. Thus, "first switch means" covers the disclosed structure, i.e., mechanical switch 22 and its structural equivalents. Specifically, Figure 2 shows mechanical switch 22 as a two-position, moveable switch. As discussed above, this court finds that the "switch moveable" element is literally present in the Intellicode. Similarly, the "first switch means," properly construed as a two-position, mechanical switch or its structural equivalent, is also literally present in the Intellicode. This court therefore affirms the district court's judgment.

"Different Codes" – Claims 1 and 5

[21] The district court also addressed the "different codes" element of claims 1 and 5, and found these elements literally present in the Intellicode. In particular, the district court construed "different codes" to mean "factory-defined codes stored within the radio transmitters which uniquely identify each different transmitter and are not selectable or modifiable by the user of the garage door opener ("GDO") system." The district court's interpretation of this term is consistent with the claim language and finds support in the written description. The claim language requires association of each code with a different transmitter. The written description notes that each code uniquely identifies a transmitter. The specification adds that the codes are set in a factory and remain unchangeable by the user "to eliminate the requirements for code selection switches in the transmitters." '364 patent, col. 1, ll. 53–54.

The claim language does not supply any further constraints on the meaning of "codes." *See Vitronics*, 90 F.3d at 1582. Claims 1 and 5 do not require an identical code in the transmitter and the receiver. The language of both claims recite "that the code of said first transmitter will be stored in said memory means." Properly construed, this language requires the memory to retain the code it associates with the "first transmitter." The claims, however, do not require that the memory store the exact sequence of coded bits transmitted from the transmitter as its identifying signal. This reading of the claim language finds support in the written description, which explains that a code is associated with a transmitter. It does not describe any particular format, encryption,

or alteration in transmitted and stored codes. The district court thus avoided interpreting "codes" so narrowly that any simple reformatting or addition of a single bit at the end of the code before storage would avoid the literal scope of claims 1 and 5.

In the Intellicode system, the unique, fixed, factory-programmed string of binary numbers in the transmitter do not match exactly the bits stored in the microprocessor's memory. Rather the Intellicode encrypts the transmitter code for transmission creating a different string of bits, the so-called "hopping code." The Intellicode microprocessor in "program" mode stores a secret key (yet another string of bits), and thereafter in "operate" mode uses the secret key to verify authorized (i.e. previously learned) transmitters. These features of the Intellicode, as explained above, however, do not avoid literal infringement of the "different codes" element. Thus, this court affirms the district court's construction of the "different codes" element and its finding that this element is literally present in the Intellicode.

III.

In conclusion, this court affirms the district court's grant of Overhead Door's motion for summary judgment of no literal infringement of claim 1, but vacates the summary judgment of non-infringement under the doctrine of equivalents. Specifically, the district court correctly found as a matter of law that the accused device literally includes the "switch moveable" and "different codes" claim elements and does not literally include the "memory selection switch" element. The district court erred, however, in deciding on summary judgment that the accused device does not contain an equivalent of the "memory selection switch." This court concludes that this is a genuine issue of material fact to be determined by a fact-finder.

As to claim 5, this court vacates the district court's grant of summary judgment of non-infringement. The district court erred in finding as a matter of law that "memory selection second switch means" is absent in the accused device either literally or under the doctrine of equivalents. This court concludes that this is a genuine issue of material fact to be determined by a fact-finder. The district correctly found, however, that as a matter of law the accused device literally contains "different codes" and "first switch means."

The court therefore remands this case to the district court for further proceedings in accordance with this opinion.

COSTS

Each party shall bear its own costs.

AFFIRMED-IN-PART, VACATED-IN-PART, and REMANDED.

Mary Rose DIEFENDERFER, Petitioner,

v.

MERIT SYSTEMS PROTECTION BOARD, Respondent.

No. 98-3404.

United States Court of Appeals, Federal Circuit.

Oct. 14, 1999.

Employee of Federal Aviation Administration (FAA) appealed decision of Merit Systems Protection Board (MSPB) dismissing her whistleblower appeal for lack of jurisdiction. The Court of Appeals, Archer, Senior Circuit Judge, held that District Court lacked jurisdiction to hear whistleblower claims by FAA employees, notwithstanding that Department of

IN THE UNITED STATES DISTRICT COURT
FOR THE NORTHERN DISTRICT OF ILLINOIS
EASTERN DIVISION

THE CHAMBERLAIN GROUP, INC., a Connecticut Corporation,	)	
Plaintiff,	)	Civil Action No. 00 CV 0454
v.	)	The Honorable Rebecca R. Pallmeyer
LYNX INDUSTRIES, INC., a Canadian Corporation, and NAPOLEON SPRING WORKS, INC., an Ohio Corporation,	)	
Defendants.	)	

Declaration of Thomas A. Gafford

1. My name is Thomas Austin Gafford. I have been retained as a technical expert in this matter on behalf of defendants, Lynx Industries, Inc. and Napoleon Spring Works, Inc. (collectively "Lynx"), by the law firm of McDonnell Boehnen Hulbert & Berghoff.

2. I am competent to testify to the matters set forth below because of my extensive experience and education in electrical engineering and in particular, digital circuit and computer software design. My curriculum vita is attached to this declaration as Exhibit 1.

3. I have read and I am familiar with U.S. Patent No. 4,750,118, U.S. Patent No. Re. 35,364 ("the '364 patent"), and U.S. Patent No. Re. 36,703 ("the '703 patent"), as well as the record of their prosecution in the U.S. Patent and Trademark Office,

including the cited prior art. I have also read the Federal Circuit's opinion in *Overhead Door Corp. v. Chamberlain Group, Inc.* 194 F 3d 1261 (Fed. Cir. 1999). In addition, I have reviewed a number of documents produced in this matter and listed in Exhibit 2.

4. I have thoroughly studied the Lynx Model LPL 1/2 HP garage door opener ("the Lynx product"), which I understand Chamberlain has accused of infringing the '364 and '703 patents. In addition, I have studied the software description, the assembly source code language and the operating manual. These are shown in Exhibit 3. The source code is programmed into a microprocessor contained in the receiver portion of the Lynx product. The microprocessor controls the operation of the garage door opener system by executing the source code. In other words, all functions of the Lynx product are dictated by the source code that is programmed into the microprocessor.

5. The Lynx product is sold with one receiver and one transmitter. The receiver portion is mounted to a garage ceiling and contains a motor that opens and closes a garage door. The transmitter is the little box that is typically kept in a homeowner's car and is used to open or close the garage door remotely by depressing a button on the transmitter. When the transmitter button is depressed, the transmitter transmits a signal in the form of a code to the receiver mounted in the garage ceiling. If the received transmitter code matches a code previously stored in the receiver's memory, then the receiver energizes the motor to either open or close the door.

6. In order for the Lynx product to operate, it must first be taught the code of at least one transmitter. The Lynx products are capable of storing up to six (6) transmitter codes. Each stored transmitter code occupies a storage location consisting of two non-volatile memory addresses. In the Lynx software, these locations are designated as 0, 1, 2, 3, 4 and 5, corresponding to address pairs 00/01, 02/03, 04/05, 06/07, 08 09, and 0A/0B.

7. In the Lynx device, when a new transmitter code is stored in the receiver, the new code is always stored in location 0. Each code that is already stored in a location is copied to the next location in order to make location 0 available for the newly added code.

8. For example, to store a transmitter code in a Lynx receiver, the receiver must be set to the program mode by depressing the program/operate button on the receiver and then depressing the button on the transmitter. This causes a first code from a first transmitter to be stored into location 0.

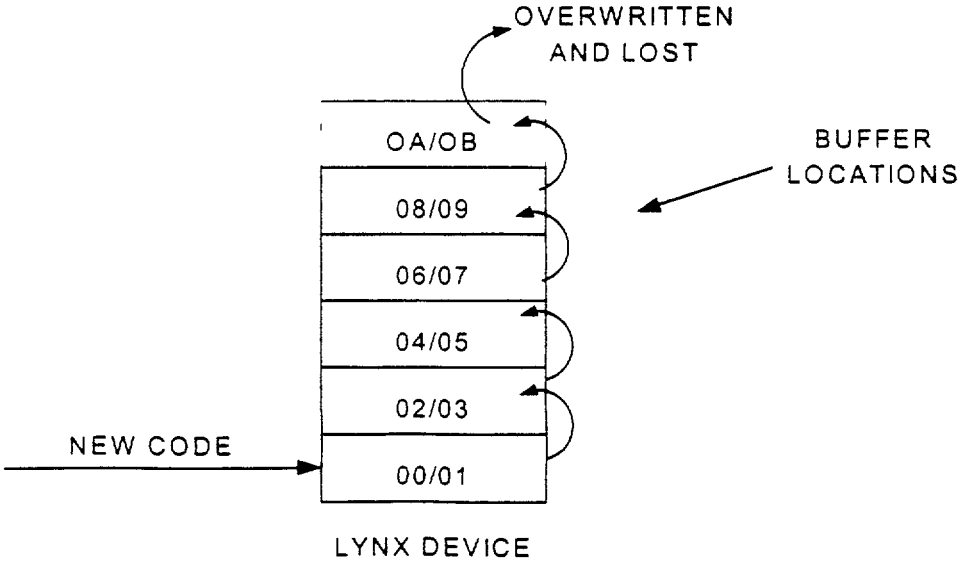
9. When a second code from a second transmitter is to be stored in the receiver, the receiver program/operate button is set to the program mode, and the button on a second transmitter is depressed. Before the second code is stored in location 0, the code from the first transmitter, presently residing in location 0, is copied to location 1, and the newly received second code from the second transmitter is then stored in the same location 0 where the first transmitter code was previously stored.

10. When a third code from a third transmitter is to be stored in the receiver, the receiver program/operate button is set to the program mode, and the button on the third transmitter is depressed. Before the third code is stored in location 0, the code from the first transmitter, that is presently in location 1, is copied to location 2. Likewise, the code from the second transmitter, that is presently in the original location 0, is copied to memory location 1. The code from the third transmitter is then stored in location 0.

11. If transmitter codes are stored in all available memory locations when a new code is received, the oldest code, stored at memory location 5, is overwritten by the second oldest code that was previously stored in memory location 4. Likewise, the code in 3 is copied into 4, the code in 2 is copied into 3, the code in 1 is copied into 2, and the code in 0 is copied into 1. Upon completion of this copying process, the newly received code is then stored in 0.

12. During the program function of storing new transmitter codes, the Lynx product can only store a newly received code in memory location 0.

13. The operation of the Lynx device is illustrated as follows:



14. A switch, whether implemented in hardware or in software, performs the function of directing an item (current, information, energy, etc.) to one place or another. The result produced by the Lynx controller of storing new codes in different memory locations is not performed using a switch. There is no decision process as to where or even whether the new code will be stored. When a new code is received in the Lynx controller, every existing code is always moved from its present memory location to a new memory location, and the new code is stored in the same memory location where every other new code was originally stored when it was first received. There is no 'switch' function performed in the Lynx controller.

15. The accused Lynx product always stores every newly received code in the same location.

16. The Lynx software does not choose from two or more alternatives for a location to store a newly received code. In other words, the software does not select where to store the newly received code, it always stores it in the same location 0.

17. None of the software on the Lynx product contains a "pointer" that remembers from one programming operation to the next where to store the next code.

18. The only software that stores a new code starts at line 231 of door.asm, and it always stores the new code starting at 00, because it clears (sets to 0) the address argument used by the non-volatile memory storage routine WRITE_EEP. The balance of

the new code is always stored in address 01, because this address is set by the INC (add one) instruction at line 224 or door.asm.

19. The Lynx product has no mechanical switch that affects where the newly received codes are stored.

20. If a code is stored in the highest memory location, indicating that all six locations are filled, then that code is lost because it is overwritten by the next to highest code when a new code is programmed.

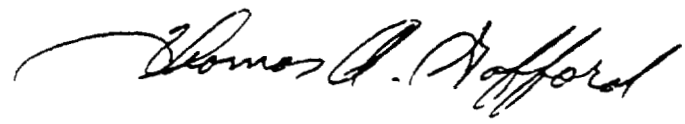
21. There is no indexing of a "pointer" to direct a newly received code to a particular memory location. Because there is no indexing, there is no checking the value of the pointer to determine whether it is equal to 5. There is no resetting of a pointer if the value equals 5. And, there is no erasing of all codes in memory when a pointer is reset.

22. When in program mode, none of the software on the Lynx product checks the newly received code to see if it already matches a code already stored. As such, there is no repetitive checking of the stored codes. When a new code is received and the unit is in program mode, the branch instruction at line 180 of door.asm jumps to STORE_PASS where the new code is stored in locations 00 and 01 without regard to the other code contents of memory.

23. Since there is no checking function to determine whether a new code entered during program mode matches an existing stored code, the same code can be entered 6 times. Therefore, it is possible to fill all available memory locations with the same code.

I declare under the penalty of perjury that the foregoing is true and correct.

Executed this 28th day of October 2000.



Thomas A. Gafford

IN THE UNITED STATES DISTRICT COURT
FOR THE NORTHERN DISTRICT OF ILLINOIS
EASTERN DIVISION

THE CHAMBERLAIN GROUP, INC., a Connecticut Corporation,	)	
Plaintiff,	)	Civil Action No. 00 CV 0454
v.	)	The Honorable Rebecca R. Pallmeyer
LYNX INDUSTRIES, INC., a Canadian Corporation, and NAPOLEON SPRING WORKS, INC., an Ohio Corporation,	)	
Defendants.	)	

Supplemental Declaration of Thomas A. Gafford

1. My name is Thomas Austin Gafford. I have been retained as a technical expert in this matter on behalf of defendants, Lynx Industries, Inc. and Napoleon Spring Works, Inc. (collectively "Lynx"), by the law firm of McDonnell Boehnen Hulbert & Berghoff.

2. I am competent to testify to the matters set forth below because of my extensive experience and education in electrical engineering and in particular, digital circuit and computer software design. My curriculum vita is attached to this declaration as Exhibit 1.

3 I have read and I am familiar with the Declaration of Dr. V. Thomas Rhyne submitted by Chamberlain in support of its motion for a preliminary injunction in this case. I incorporate herein by reference my earlier declaration dated October 28, 2000, which I provided to support Lynx's motion for summary judgment of non-infringement.

4. I completely disagree with Mr. Rhyne's statements that the Lynx products have any type of switch that is capable of selecting a memory location to store newly received transmitter codes. Indeed, as my earlier declaration states, the Lynx product is designed to store each newly received code in the exact same memory location as each

previously received code. There is no software means or mechanical device that "selects" or "switches" where the newly received codes are stored.

5. As to claim 29 of the '703 patent, if the recited "software controlled memory selector" is to have a meaning, it must be as described by Dr Rhyne in his declaration found in the '703 patent file history, where in para. 11 he makes clear that the patent specification "refers to the software memory selection switch as a "code location pointer". He goes on to say that "code" is the new transmitter identity code, and "location" is the memory address at which that (new) code is stored. Thus the recited "selector" must be synonymous with "switch," it must function as such and is not found in the Lynx controller.

6. The Lynx software does not choose from two or more alternatives for a location to store a newly received code. In other words, the software does not select where to store the newly received code; it always stores it in the same location 0.

7. I also disagree with Mr. Rhyne's conclusion that the Lynx product has a code location pointer. None of the software on the Lynx product contains a "pointer" that remembers from one programming operation to the next where to store the next code.

8. Mr. Rhyne concludes that the memory location EEP_ADDR in the Lynx software source code is the equivalent of the memory selection switch required by the Chamberlain patents. I disagree with that conclusion. The EEP_ADDR is not a "code location pointer" as defined by Chamberlain's patent, in particular as set forth in Fig. 3 of the patent specification. The EEP_ADDR in the Lynx source code is a "variable" that constantly changes value depending on which subroutine program is using it. The EEP_ADDR variable is akin to a roadway intersection – data is passed through the intersection but nothing is stored in the intersection.

9. In order for a software pointer to function as a switch, it must be stored in computer memory, it must be alterable to have different settings, like a mechanical switch, and it must remain in the last position it was set for storing a code until it is changed by the software for storing another code. This is how the "code location pointer" operates in the Fig. 3 flowchart of the '703 patent, and is consistent with Dr. Rhyne's file history declaration in paragraph 22 where he points out that, for a memory value to be a pointer,

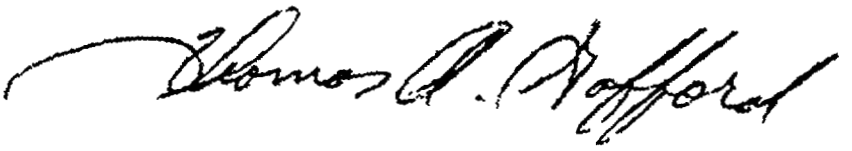
it must be maintained in a memory location of its own. The variable EEP_ADR is not such a pointer. It is not dedicated to storing codes in particular addresses. It is, in fact, merely a way for the software to pass an address value to the memory writing and reading routine, and it is used for other purposes besides specifying a code write address. Specifically, it is also used to control clearing of the memory, reading memory to find a matching code, and for reading and writing the door motion limit information. When it is being used for writing a new code, it is always set to zero by the permanently set and unchanging new code write pointer, which has the value of zero, that is inherent in the "CLR" instruction found 8 lines from the end of page 5 of the software document Door.asm. (Bates number LX1150).

10. The only software that stores a new code starts at line 231 of door.asm, and it always stores the new code starting at 00, because it clears (sets to 0) the address argument used by the non-volatile memory storage routine WRITE_EEP. The balance of the new code is always stored in address 01, because this address is set by the INC (add one) instruction at line 224 or door.asm.

11. A switch, whether implemented in hardware or in software, performs the function of directing an item (current, information, energy, etc.) to one place or another. The result produced by the Lynx controller of storing codes received at different times in different memory locations is not performed with any kind of switch. When a new code is received in the Lynx controller, every existing code is always moved from its present storage location to one and only one new location, not one of two new locations, and the new code is stored in the same location as every other new code, not one of two or more locations in memory There is no A/B, either/or, or any other 'switch' function performed in the Lynx controller. There is no decision process as to where or even whether the new code will be stored. The Lynx controller uses a completely different way to produce the result of storing in different locations codes received at different times, than the way disclosed and claimed in the '364 and '703 patents.

I declare under the penalty of perjury that the foregoing is true and correct.

Executed this 21st day of November 2000.



Thomas A. Gafford

Thomas A. Gafford
553 North Pacific Coast Highway PMB235
Redondo Beach California 90277
Phone: (310)316-3963 • Fax: (310)316-3884
email:tag@gafford.com

Curriculum Vitae

Education

University of Washington, Seattle, Washington
Bachelor of Science in Electrical Engineering. 1972

Stanford University, Palo Alto, California
Attended Master of Science in Electrical Engineering program. Coursework included logic and circuit design, LISP, software algorithm design and system programming 1972 - 1973.

Professional Societies

Institute of Electrical and Electronic Engineers, member.

Experience

Gafford Technology, Redondo Beach, California
Owner. 1986 to present.
Presently, designing and manufacturing computer peripheral switches using high level software design language, consulting in computer system design, software selection and network configuration, and as a technical expert in patent and other litigation matters involving hardware and software systems. Specific tasks have included analysis of small to large scale software and hardware systems for their relationship to specific patent claims, trade secrets, copyrighted material and electronic evidence discovery. Awarded three patents on SCSI bus switching equipment. References on request.

Softix Inc., Campbell, California
Engineer. 1983 - 1986
Responsible for selection, evaluation, integration, installation, customer staff training and repair of all hardware components of a mixed - vendor dual minicomputer system. Latest product was implemented in a UNIX environment using the C language. Responsibilities also included management of a software development team, large scale software system architecture and design, sales, contracting, production, and field support, including software analysis, debugging and writing programs for feature enhancement. Solely responsible for operating system driver program development and enhancement. The systems are used for control and sale of entertainment tickets by ticket agencies and large arena complexes. Projects involved work in Canada, Australia, Hong Kong, and major cities in the United States. Designed software architecture and electronics, and supervised

software and mechanical engineering for a graphic ink jet ticket printer having multiple microprocessors.

G Systems, Santa Clara, California
Owner. 1976 - 1983
Same duties as above, as well as the following:
Designed software interface and hardware of message - oriented disk controller for DEC PDP11 computers at Microcomputer Systems (Sunnyvale, California). Assisted design of emulator for DEC PDP10 computer at Foonly, Inc. (Mountain View, California). Assisted legal team as technical expert in litigation between DEC and Microcomputer Systems over patents involving MASSBUS equipment. Conceived and managed hardware and software design and development of microcomputer - controlled ticket printer for BASS Tickets (Oakland, California), with emphasis on reliability and maintainability. Design focused on replacing hard-wired functions in previous units with software functionality. Responsible for selection, evaluation, integration, installation, staff training, and repair of all hardware components and some software of a mixed - vendor dual minicomputer system. Duties included software debugging and writing enhancements in the field. The systems are used for control and sale of entertainment tickets by ticket agencies and large arena complexes. Wrote communications software and device driver for transportation ticketing system.

Stanford University, Palo Alto, California
Engineer. 1973 - 1976
Responsible for computer engineering projects in association with robotics research group at Stanford Artificial Intelligence Laboratory. Duties included design, construction and debugging DC motor control and sensor electronics for a robotic arm and their interfaces to a DEC PDP11 computer, and maintenance and enhancement of a large assembly-language CAD software system used for logic design and PC board layout.

Other relevant experience:
ca. 1968, taught basic electronics in on-the-job training program, USAF.

1955-1971, analysis, repair, and modification of equipment ranging from home entertainment devices to large-scale digital computer systems.

Thomas A. Gafford

Experience assisting clients and their litigation teams (*** marks cases in which I have testified at trial):

1/00 to 5/00, assisted **Fonar** in patent litigation involving the electronics of magnetic resonance imaging equipment brought against **Healthsouth**.

9/99 to 5/00, assisted **Midtronics** in patent litigation brought against **Actron** relating to electronic control of automobile battery testing equipment.

7/99 to present, assisting **HCL** in software contract litigation brought by **GEAC** relating to completion and quality of a large commercial software project.

12/98 to 3/00, assisted **Xerox** in patent litigation brought against **Hewlett-Packard** in matters relating to electrical circuitry of thermal inkjet printheads.

7/98 to 8/98, assisted **DSL Transportation** in contract litigation brought by **Interactive Systems Inc.** in matters relating to software development practices.

5/98 to present, assisting **V-Communications, Inc.** in patent litigation brought by **Power Quest** in matters relating to disk partitioning software.

*** 5/98 to 7/99, assisted **ETM Entertainment Ticketing Network** in trade secret litigation brought by **Retrac** in matters relating to transaction processing software.

1/98 to 1/99, assisted **Robertson Machine** in trade secret litigation brought against **Serpa** in matters relating to NC programming information.

1/98 to 8/98, assisted **Emerson Electric** in patent litigation brought against **Quorum International** in technology relating to AC motor control.

*** 1/98 to 11/98, assisted **Honeywell** in patent litigation brought against **Foxboro** in technology relating to industrial process control sensor systems.

7/97 to 3/99, assisted **High Performance Appliances** in patent litigation brought by **Worldtronics, Inc.** In technology relating to appliance timing and control.

6/97 to present, assisting **United Technologies Automotive** in patent litigation brought by **Fulhorst** in technology relating to automobile alarms.

6/97, assisted **Harris Semiconductor** in patent litigation brought against **Hyundai** in technology relating to static ram IC circuitry.

6/97 to 1/98, assisted **Card-Monroe Corp.** in patent litigation brought against **Tuftco** in technology for carpet tufting machine controls.

5/97 to 9/98, assisted **Raychem** in patent litigation brought by **Oneac** in technology for telephone line surge suppression.

4/97 to 7/98, assisted **Acer** in trade secret litigation brought by **Zeny** and **Henson Mou** in technology of PC motherboard design.

4/97 to 12/97, assisted **Matra Enterprise Co. Ltd.** (Taiwan) in contract litigation brought against **IBM** in technology relating to PC motherboard design and manufacturing.

2/97 to 11/98, assisted **Honeywell** in patent and contract litigation brought by **Foxboro** in technology relating to industrial process control communication systems.

11/96 to 12/97, assisted **United Technologies Automotive** in patent litigation brought against **National Semiconductor** in technology relating to power control integrated circuits.

*** 6/96 to 8/00, assisted **IMS / Hurco** in patent litigation brought against various defendants in technology relating to machine tool controls.

6/96 to 12/97, assisted **Crane Packing Co.** in litigation brought against **Durametallic** and **Josef Sedy** involving gas seal analysis software.

11/95 to 3/96, assisted **Fritz Forwarding** in litigation brought by **Waffer** involving value of computer monitors based on service history and construction quality.

8/95 to 8/98, assisted **Los Angeles Department of Water and Power** in litigation brought by **American Computer Manufacturing** relating to the value of used computer equipment.

7/95 to present, assisting **Quantum Group** in patent litigation brought against **American Sensors, Nighthawk, and Xintex** involving carbon monoxide detector technology.

7/95 to present, assisting **Dictaphone** in patent litigation brought by **Sudbury Systems** involving voice storage and retrieval technology.

*** 7/95 to 9/96, assisted **Global Gaming, Inc.**, in patent litigation brought against **IGT** involving computerized slot machine technology.

7/95 to 1/96, assisted **Fairplan Insurance Co.** in litigation brought by **Alevy** relating to the value of appliance monitoring computers.

6/95 to 3/96, assisted **MTI** in patent litigation brought against **CMD** and **Falcon** involving RAID storage technology.

3/95 to 10/95, assisted **Mallory** in litigation relating to remote sensing of fluid storage and delivery systems brought by **Tidel**.

3/95 to 11/95, assisted **Black & Decker** in patent litigation relating to appliance timing and control brought by **Worldtronics**.

12/94 to 2/95, assisted **Midtronics** in patent litigation involving electronics for battery chargers brought against **Telecom Assistance Group**.

10/94 to 4/96, assisted **Rockwell** in patent litigation relating to telephone announcement systems brought by **Golden Enterprises**.

*** 5/94 to 5/95, assisted **Fonar** in patent litigation involving the electronics of magnetic resonance imaging equipment brought against **General Electric** and **Hitachi**.

11/92 to 4/95, assisted **Mitsubishi Corp.** in patent litigation relating to machine vision and computer information storage and retrieval brought by **Jerome Lemelson**.

6/94 to 1/95, assisted **AST** in patent litigation involving technology for personal computers against **Texas Instruments**.

*** 4/94 to 1/95, assisted **A & L Technologies** in patent litigation involving electronics for hearing aid amplifiers brought against **Resound**.

7/94 to 12/94, assisted **AT&T** in trade secret litigation involving electronics of voice prompting systems brought by **Golden Enterprises**.

3/93 to 12/94, assisted **Apple Computer** in patent litigation relating to image storage and retrieval brought by **Jerome Lemelson**.

11/93 to 1/94, assisted **FMC Gold** in trade secret litigation relating to image processing and database analysis systems brought against **Russel Cowart**.

8/93 to 6/94, assisted **Maxtor** in patent litigation relating to disk drive electronics brought against **NEC**.

3/93 to 12/94, assisted **Qantel** in trade secret litigation relating to operating system, communication protocol design, and compiler development brought against **Signature Systems** and against **Texas Instruments, Inc.**

*** 10/93 to 9/94, assisted **Octel** in patent litigation relating to voice message storage and retrieval brought by **Theis Research**.

*** 1/94 to 7/94, assisted **Mitsubishi Electronics** in patent litigation relating to memory modules brought by **Wang Laboratories**.

2/93 to 8/94, assisted **Lawrence Lockwood** in patent infringement case involving transaction processing systems and terminals brought against **American Airlines**.

1/93 to 1/94, assisted **PCC, Inc.** in discovery matters relating to computer media in litigation brought by **California Hospitals Association**.

3/92 to 12/92, assisted **Seatt Corp.** in patent infringement case involving microcomputer controlled thermostats brought by **Honeywell**.

1/92 to 11/92, assisted **Kapali Eswaran** in patent infringement case involving microcomputer controlled telephone answering machines brought against **AT&T**.

1/92 to 5/92, assisted **Qume** in trade secret case involving microcomputer disk drive controllers brought against **Ultra-Stor**.

10/91 to 7/92, assisted **Banner Aeronautical** in defense of wrongful termination action brought by **Christianne Maho**.

8/91 to 7/92, assisted **Ventritex** in defense of trade secret suit brought by **Intermedics**, involving analog and digital circuitry used in pacemaker and defibrillation devices.

8/91 to 2/93, assisted **Daewoo** in patent infringement case involving personal computer and microprocessor technology brought by **Texas Instruments**.

7/91 to 1/93, assisted **IGT, Inc.** in patent infringement case involving microcomputer controlled gaming machines brought against **Bally Corp**.

11/90 to 3/91, assisted **Western Graphtec, Inc.** in defense of patent infringement suit in the area of thermal printing chart recorders brought by **Astro Med, Inc.**

9/90 to 2/93, assisted **Telepanel, Inc.** in patent infringement case brought against **Pricelink, Inc.** and against **ERS**, involving retail shelf price electronic display systems. Responsible for circuit and microprocessor software issues.

2/90 to 1/93, assisted **FMC** corporation in trade secret case involving former employee of FMC, and **Hennessy** company. Technology is automotive wheel service equipment and associated electronics and optics. Responsible for circuit and microprocessor software issues.

*** 6/89 to 9/90, assisted **Motorola**, Inc. in patent and contract case against **Hitachi**, Ltd. of Japan, involving analysis of MOS integrated circuit related patents and contract disputes centered around comparisons of software and hardware characteristics of microprocessor components. Responsible for coaching industry leaders as testifying experts in issues involving microprocessor architecture and circuit design.

*** 2/89 to 11/92, assisted **criminal** defendant in matters related to destruction of defendant's data kept on IBM-compatible PC disk system. Work includes analysis of data structure of all areas of disk seized by authorities.

2/88 to 10/89, further work with **IJR** attorneys, in patent infringement suit against **Sodick** company of Japan and their U.S. subsidiary (same technology as below, with emphasis on extensive analysis of software involved in patent infringement).

*** 3/88 to 10/88, assisted **Westek** of San Diego, CA, in patent infringement suit against **Tri-Lite** of Chicago, IL, to analyze circuits involved in lamp dimmer modules. Managed creation of animated model of patented invention and accused infringing circuit. Responsible for developing case position relating to transistor circuitry issues, and for testifying.

6/87 to 1/90, assisted **Business Records Corporation** to analyze prior art patents in the area of electronic voting machines to determine whether patents held by **BRC** and by **Shoup** Corporation were valid and whether equipment made by either company infringes the other's patents. This analysis entails understanding low level analog electronics and standard and semi-custom LSI digital circuitry and software.

11/85 to 1/87, assisted **IJR Ltd.** of Japan, to analyze prior art patents and schematic drawings of EDM (Electrical Discharge Machining) equipment to determine whether patents held by IJR were valid, and analyzed schematics of EDM equipment sold by **Colt Industries' Elox** division to determine if that equipment infringed the IJR patents. This analysis entailed understanding of low and high-level analogue electronics circuitry, as well as computer circuitry and software.

*** 1/81 to 2/82, assisted **Microcomputer Systems** in Sunnyvale Calif., to analyze prior art consisting of logic schematics of computer equipment made by IBM and **DEC**, to determine whether they invalidated patents held by DEC on their MASSBUS. This analysis required understanding of several different families of digital logic and styles of schematic drawings.

Listing of Materials Reviewed In Preparation of My Declaration

1. Assembly source code for the for Lynx products.

2. Chapter 2 of Leventhal's book "Introduction to Microprocessors: Software, Hardware, Programming

3. Introduction to Klingman's book: Microprocessor Systems Design

4. Installation and operation manual, advertising sheets, flow diagrams, circuit diagrams, computer chip descriptions for the Lynx Proline garage door opener – Production Nos. LX1086-LX1136

5. Chamberlain's motion for preliminary judgment against Lynx.

6. Federal Circuit opinion in *Overhead Door Corp. v. Chamberlain Group, Inc.* 194 F.3d 1261 (Fed. Cir. 1999).

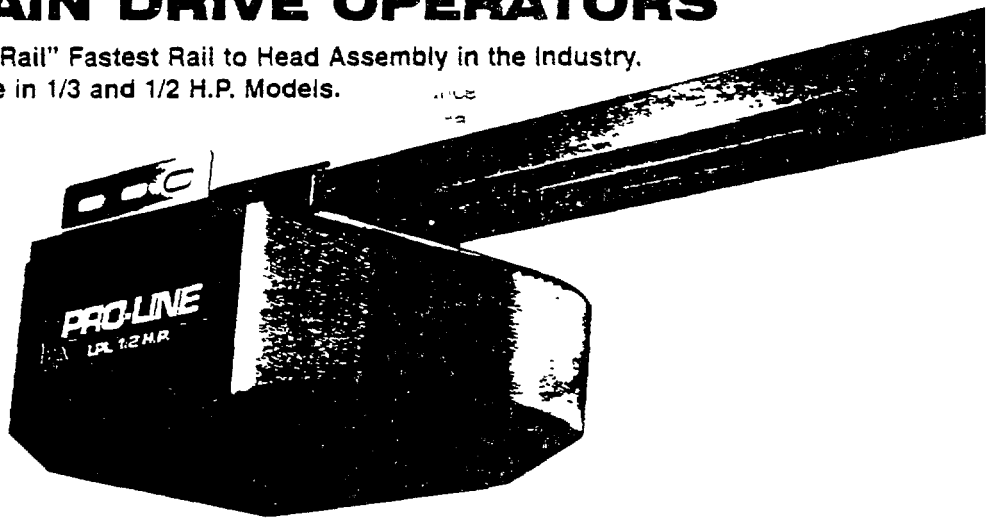
Serving Door Installers for Over 25 Years

Lynx PRO-LINE

Exclusively Sold through the Professional Garage Door Industry

CHAIN DRIVE OPERATORS

"Rapid L Rail" Fastest Rail to Head Assembly in the Industry.
Available in 1/3 and 1/2 H.P. Models.



FEATURE

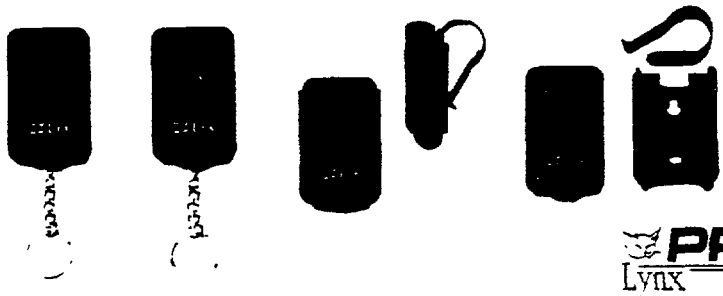
- Designed to Last Twice as Long as Some Operators
- Rapid "L Rails" for Maximum Strength
- Non-Polarity Sensitive Photo Eyes
- Ultra-Secure Radio Controls
- Up to 195 Pounds of Lifting Force
- Industry Standard Photo Eye Wiring
- Self-Diagnostic Electronics
- Service Friendly
- Up Down Stop Logic in Both Directions
- Automatically Locks Door in Closed Position
- Automatic Thermal Protected Motor

BENEFIT

- A Better Value to Sell Your Customer
- Quickest and Easiest Rail to Head Assembly
- No Color Code Wires or Terminals
- Forget about Code Grabbers
- Strong Enough to Handle the Toughest Door
- Works in all Pre-Wired Garages
- Helps Eliminate Costly Callbacks
- All Internal Wires Have Connectors
- Easy Helical Gear Replacement

ULTRA SECURE MS TRANSMITTERS

Industry first and only mid-size transmitters that can either be mounted to your visor, wall or carried on a key ring.



PRO-LINE™
Lynx

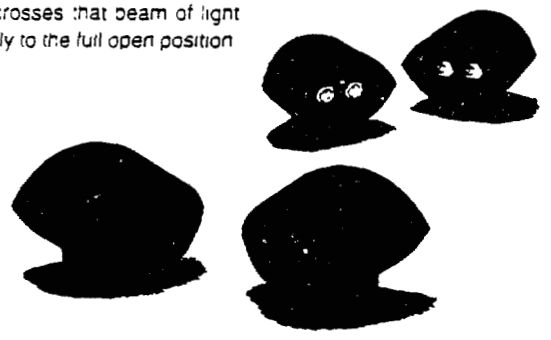
LX 1086

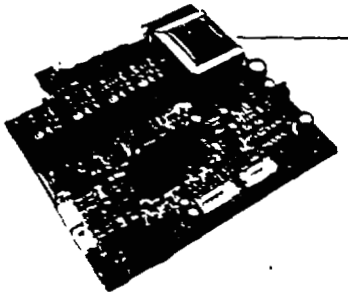
Lynx PRO-LINE

SILENT GUARD PHOTO EYE SYSTEM

Silent Guard non-contact reversing system transmits an invisible light across the inside of the garage door opening. If anyone or anything crosses that beam of light while the door is closing, the opener will reverse automatically to the full open position

- Universal industry standard wiring
- Non-polarity sensitive - color coded wires and terminals are no longer required
- High resolution optics for easy alignment and dependable operation
- Weather and moisture resistant Photo Eye case
- Compact case design

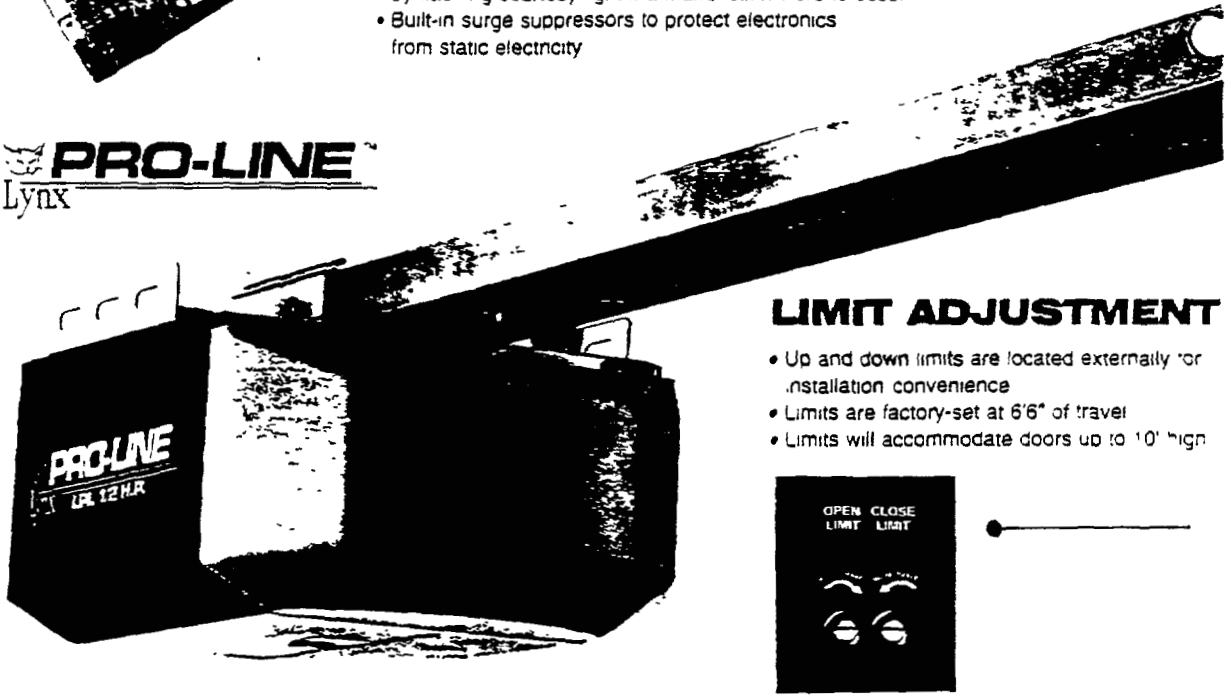




LOGIC BOARD

- Electronics with state-of-the-art design for years of trouble-free operation
- Solid-state courtesy light turns on when unit is activated and turns off automatically 4½ minutes later
- Self-diagnostic system constantly monitors operations and communicates by flashing courtesy light if a malfunction were to occur
- Built-in surge suppressors to protect electronics from static electricity

Lynx PRO-LINE



LIMIT ADJUSTMENT

- Up and down limits are located externally for installation convenience
- Limits are factory-set at 6'6" of travel
- Limits will accommodate doors up to 10' high

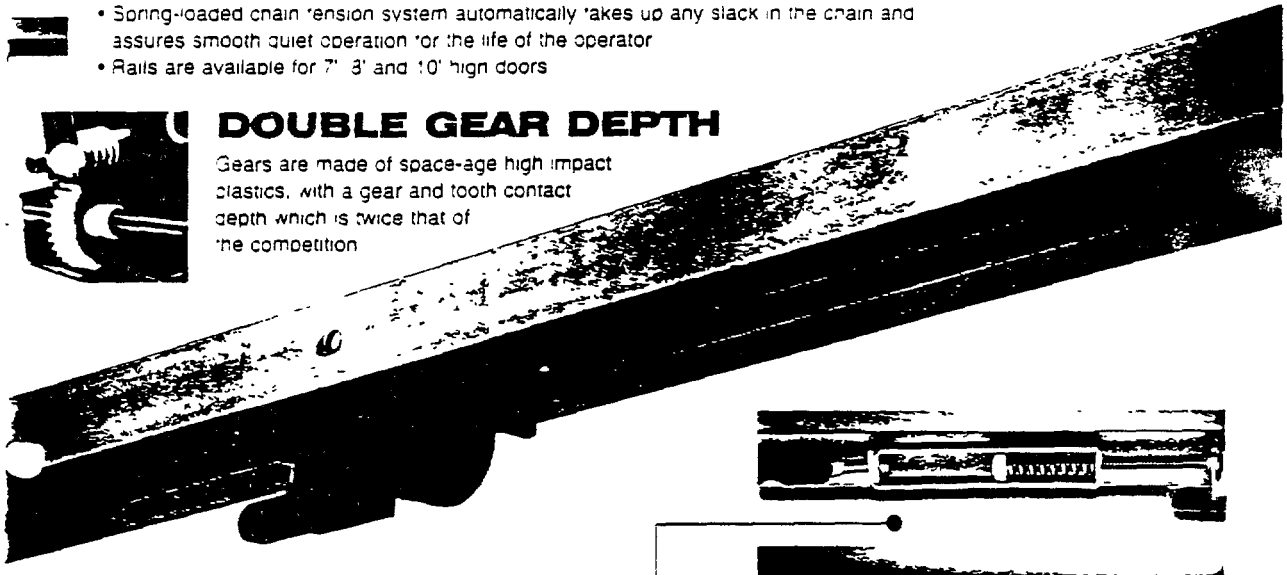
Available in 1/3 H.P. Model LPL 33, and 1/2 H.P. Model LPL 50

LX 1087

AUTO-RECONNECT SELF-LUBRICATING TROLLEY

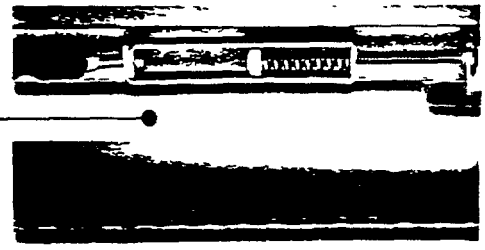
In a power failure or any time manual operation is needed, the trolley can be disconnected just by pulling on the release handle. To go back to automatic operation, just pull the handle towards the power head and activate the operator. The operator will automatically reconnect the trolley.

- Auto-reconnect trolley is made of high impact fiber filled nylon for years of trouble-free service
- High impact nylon trolley is self-lubricating for years of maintenance-free operation
- Spring-loaded chain tension system automatically takes up any slack in the chain and assures smooth quiet operation for the life of the operator
- Rails are available for 7', 8' and 10' high doors



DOUBLE GEAR DEPTH

Gears are made of space-age high impact plastics, with a gear and tooth contact depth which is twice that of the competition

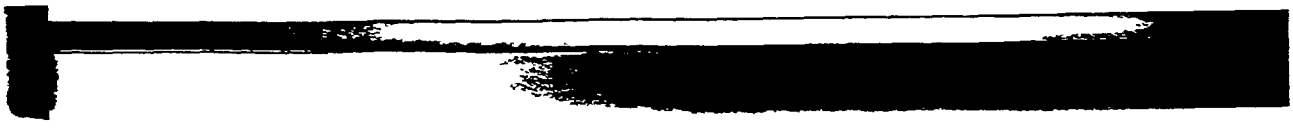


SPRING LOADED CHAIN TENSION ADJUSTMENT

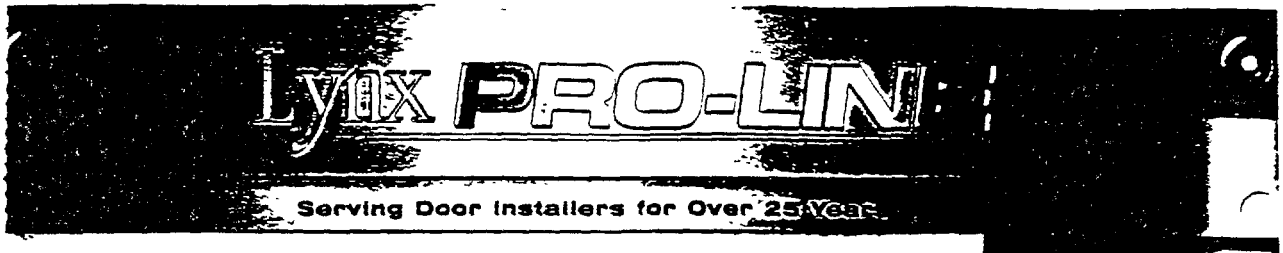
- No tools are required, just hand tighten the chain adjustment bracket
- Spring will help maintain chain tension, reducing costly callbacks
- Spring will absorb sudden chain impact, creating a smooth, quite operation

RADIO CONTROL AND OBSTACLE SENSING PROGRAMMING

- Electronically programmed internal obstacle sensing has separate up and down adjustments to provide maximum flexibility for every application
- Obstacle programming has 12 levels of adjustment to accommodate the lightest to heaviest garage doors
- Up to 6 ultra-secure MS or WKP (wireless keypad) transmitters can be programmed to each receiver
- Over one million factory-set codes give the ultra-secure radios virtually zero chance of duplication or code grabbing

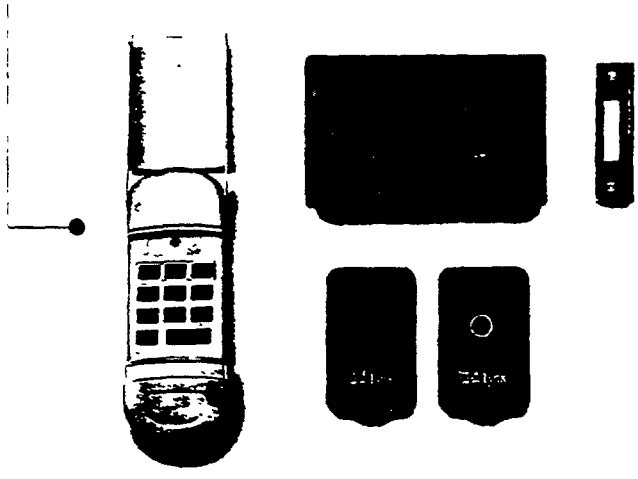


LX 1088



OPTIONAL ACCESSORIES

- Wireless keypad with cover can be programmed with codes from 1 to 6 digits, for maximum security. Includes back-light for your convenience when used in poor lighting situations
- 2-channel transmitter operates 2 garage door operators independently
- 1-channel transmitter can be used in second car to activate doublewide doors
- Deluxe 3-function wall station, turns courtesy light off/on, activates operator, and vacation switch turns off radio signal



PRO-LINE™
Lynx

ARIZONA
01 North 57th Avenue
Phoenix, Arizona, 85043
Toll free: (800) 233 • 3599
Tel. 602) 352 • 1010
Fax. 602) 352 • 1341

CALIFORNIA
5801 Stagg Street
Van Nuys, California, 91406
Toll free: (800) 369 • 6351
Tel. 818) 908 • 9935
Fax. (818) 908 • 1151

NEW JERSEY
56 Gould Avenue,
Paterson, New Jersey 07533
Toll free: (800) 234 • 5969
Tel. 973) 278 • 5588
Fax. 973) 742 • 9832

ONTARIO
3040 Mavis Side Drive, Mississauga
Ontario, Canada L5T-1J4
Toll free: (888) 839 • 5969
Tel. 416) 674 • 4606
Fax. 905) 670 • 0806

OHIO
111 Weres Drive
Archbold, OH 43502
Toll free: (800) 338 • 5399
Tel. 419) 445 • 1010
Fax. (419) 448 • 2618

QUEBEC
175 Jober Edson St-Lambert
Quebec, Canada J4R-2R3
Toll free: (888) 589 • 5969
Tel. 514) 866 • 1068
Fax. 450) 466 • 9852

FCC Certified, DOC Certified

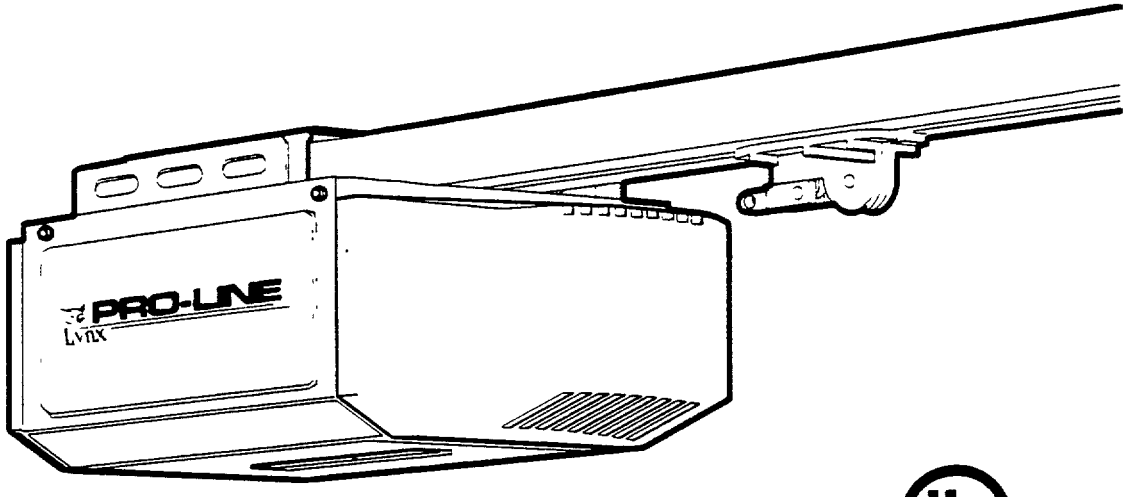
IDA DASMA

Copyright 1999 Lynx Industries. All rights reserved. PRO-LINE is a trade-mark of Lynx Industries

Dealer Brochure-1
Lynx Industries October 1999

PRO-LINE
Lynx

MODEL LPL 33 and LPL 50
Residential Garage Door Opener
for Sectional Garage Doors up to 10' high



c UL US LISTED

Installation Instructions
and
Owner's Manual

READ THIS MANUAL CAREFULLY BEFORE BEGINNING INSTALLATION

Installer: Place this manual in the plastic envelope provided and permanently attach to the wall.

LX 1090

Contents

Safety Instructions		03
Before Starting		03
Pre-Assembly Check		03
Product Features/ Specifications		04
Door Test		05
Tools Required		05
Component Identification & Inventory		06
Fastener Identification		06
Assembly Instructions		07
Installation Notes		08
Step 1	Mounting the Front Bracket	08
Step 2	Attach the Rail Assembly to Front Bracket	09
Step 3	Position the Power Head for Mounting Bracket	09
Step 4	Mounting the Power Head to Ceiling	09
Step 5	Recheck all Fasteners for Tightness	09
Step 6	Door Arm, Bracket and Plate Installation	10
Step 7	Check Emergency Release	10
Step 8	Installing the Push Button	11
Step 9	Install the Bulb	11
Step 10	Connect to Power	11
Step 11	Setting / Changing Transmitter Codes	11
Step 12	Installation of Photoeyesystem	12
Final Alignment and Test		13
Operation and Adjustment Instructions		14
Setting the Limits		14
Setting the Sensitivity Force		15
Testing the Openers Reversing System		16
Testing the Photoeyesystem		16
Final installation		17
Trouble Shooting		18
After the Installation		19
Parts Breakdown		20
Warranty		22

NOTICE

All Model LPL 33 and LPL 50 Garage Door Openers are designed and tested to offer safe operation. Provided installation and use of this product is followed in strict accordance with these instructions for assembly and installation. Failure to comply with these instructions could cause property and/or bodily injury. The opener is intended only for the use described in this manual and use other than intended will void any and all warranties herein described.

Read and Follow These Important Safety Instructions

You will see **Warning** and **Caution** statements on the following Pages
Read and follow these safety instructions carefully Failure to do so could result in serious personal injury or death

Warning means that severe injury or death may result from failure to follow instructions

Caution means that property damage or injury may result from failure to follow instructions

- Be sure to read and follow all instructions carefully
- To reduce the risk of electric shock, this equipment has a grounding type plug, that has a third (grounding) pin This plug will only fit into a grounding type outlet. If the Plug does not fit into the outlet, contact a qualified electrician to install the proper outlet. Do not change the plug in any way.
- Check to make sure the garage door is properly installed and balanced. Because of the extreme tension most garage door parts are under do not attempt to adjust on your own Have a qualified garage door service person make repairs to cables, spring and other hardware before installing the opener
- Install Entrapment Warning Label next to control button Read the Control Adjustment Warning Label Install Emergency Release Tag to the Emergency Release Cord. Mount the Emergency Release Knob 6 feet from the floor. Use the manual release Only to disengage the trolley. Do not use the Red Release Cord and Knob to pull door up or down. If possible, use the Emergency Release only when the door is closed.
- Do not connect opener to power source until instructed Install door opener 7 feet or more above floor
- After installing opener, the door must reverse when it comes in contact with a 1-1/2" high object Check this safety feature often.
- Remove all ropes connected to the garage door.
- Disengage all existing garage door locks to avoid damage to the garage door.
- Fiberglass, Aluminum and Steel Doors must be Reinforced to Prevent Damage. Consult with Manufacturer for Recommendations
- All installation and wiring must be done in strict compliance with local and state building and electrical codes Connect the power cord to a properly grounded outlet only Do not in any way alter or remove the grounding pin Photo Electrical Eyes must be installed properly. Opening doors must not close and Closing doors must open. See test procedure page 12 and page 13.
- Locate lighted push button within sight of garage door, away from all moving parts and out of reach of children (minimum 5' above floor). To reduce the risk of injury to persons, to use this operator only with a sectional door
- Never operate the opener if the system is not operating properly
- Always disconnect electric power before making repairs or removing cover.
- Activate opener only when the door is in full view and free from obstructions.
- No one should enter or leave the garage while the door is moving. Do not allow children to play near, or operate the door Keep the remote control away from children.
- After Installation is complete fasten this manual near the garage door. Perform periodic safety checks and recommended maintenance and adjustments.

Product Features

1 Motor Permanently Lubricated, thermally protected, heavy duty motor with Automatic reset.

2. Opener Lights Turn on and off automatically with 4-1/2 minutes illumination for your safety and convenience

3 Sensing System A built-in sensing system detects obstructions during door operation. If in the Downward (close) travel mode, the Opener will sense an obstruction and reverse to the full open position In the Open mode, the Opener will stop In both cases the light will start flashing and continue to flash for 60 seconds Since all doors are different, the Sensing System has independent adjustments for customizing the level of force for the normal opening and closing of specific door

4 Close Limit Switch. In winter months it's common for small pieces of ice or packed snow to be trapped under the door. Ground swelling may also effect the close limit setting of the Opener The Close Limit Switch overrides the Sensing System under the last one (1) inch of closing travel and prevents the door from reversing if it encounters an obstruction at this point.

5. Emergency Release. A pull cord allows manual disconnect and operation of door during power failure. Unit will automatically reconnect when release is reset (the trolley release lever is snapped back to its original position), power is restored and Opener is activated.

6. Mechanical Door Lock: When properly adjusted, opener locks door in closed position preventing unwanted entry

7. Easy Connect Continuous Monitor Entrapment System: System allows quick and easy installation of "Silent Guard" Photoeyesystem while control circuitry monitors these devices continuously for proper operation.

8. Constant Contact to Close: for utmost safety if "Silent Guard" Photoeyesystem fails constant contact of mechanical push button is necessary to close door. In this mode of operation, a radio transmitter cannot be used to close door

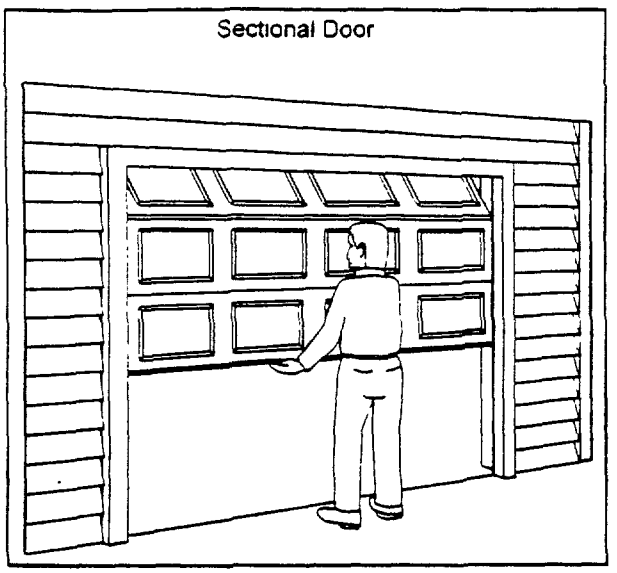
9. Momentary Contact to Close: Single touch to Radio Transmitter or Wall Button will allow door to close as long as Silent Guard Photoeyesystem is operational.

10. Silent Guard Photoeyesystem: An invisible infra-red beam of light guards the door opening and reverses a downward moving door if the beam is broken by a stationary or moving object. If the beam is broken, the opener light will flash for 60 seconds. Motor control circuitry constantly monitors the Silent Guard Photoeyesystem for proper operation.

11. Digital Radio Control: Built in allowing over 1 6 million private codes, easily selected without use of tools. Bright transmitter LED indicates operation and monitors battery condition.

Door Test

Before beginning installation of the operator please complete the following test to insure that your door is balanced and in good working condition A poorly balanced door could cause severe personal injury and damage to the opener Always have a qualified garage door service person make any needed adjustments and/or repairs to your door before proceeding with installation.



1 Raise and lower the door and check closely for areas of sticking and binding. Check for loose hinges, wobbly rollers, frayed cables and damaged or broken springs Contact a qualified garage door service person to make the necessary adjustments.

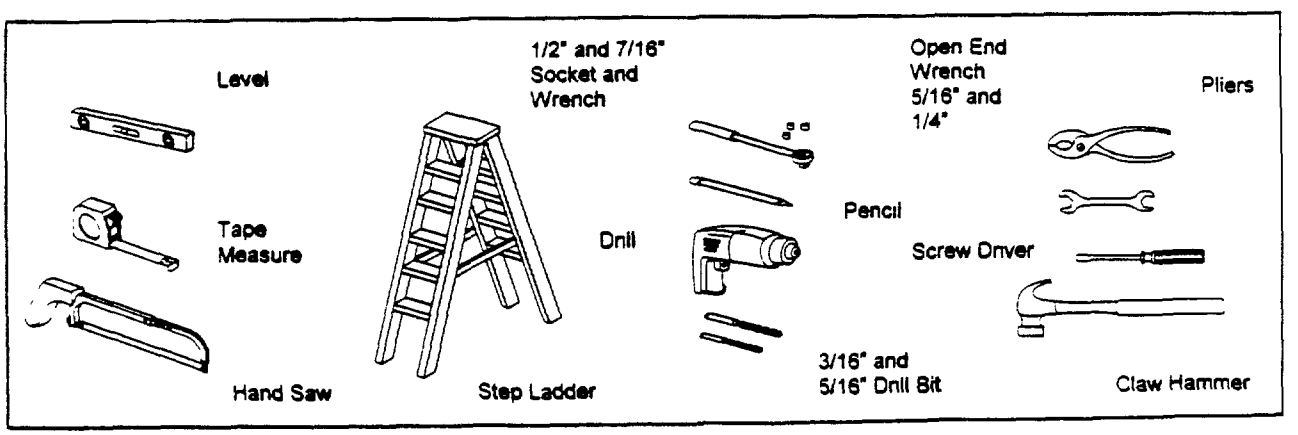
2. Lift the door approximately halfway When released, the door should stay in that position If door pulls open or moves downward, the spring mechanism is not adjusted properly Contact a qualified garage door service person to make the necessary adjustments

3. When properly installed and adjusted the door will remain clear of the opening, when allowed to rest at its natural full open position If door drifts up or down the door is not adjusted properly. Contact a qualified garage door service person to make the necessary adjustments.

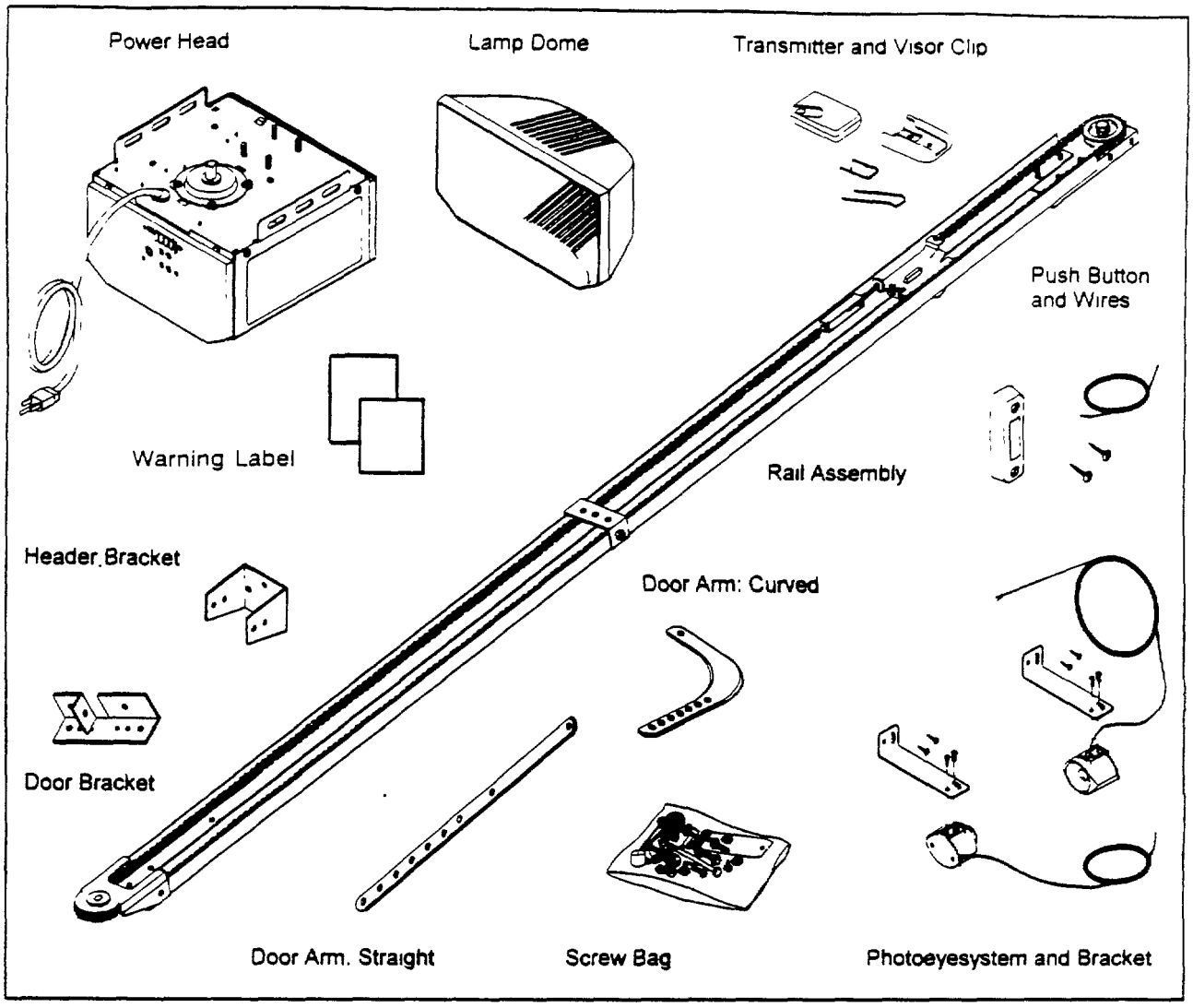
Do not install the opener until these adjustments and repairs have been made.

Carefully follow the instructions for the assembly and installation of the garage door opener contained in this manual

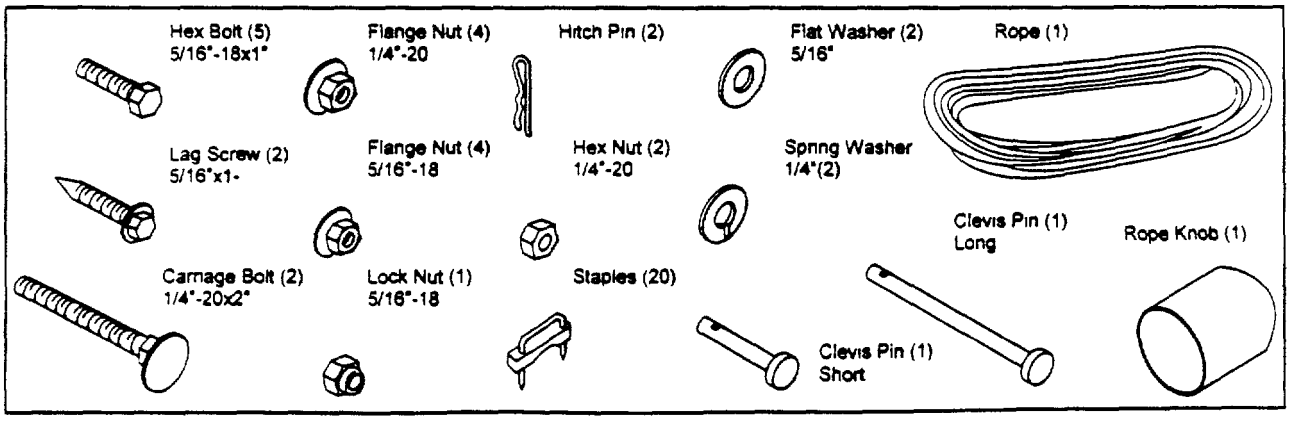
Tools Required for Assembly and Installation



Components Identification and Inventory



Fastener (parts in screw bag) Identification



Note: The L-Rail/Chain Speed Unit Assembly with Trolley and Front Idler Sprocket is packaged separately from the Power Head Unit Refer to Page 6 for package contents and identification.

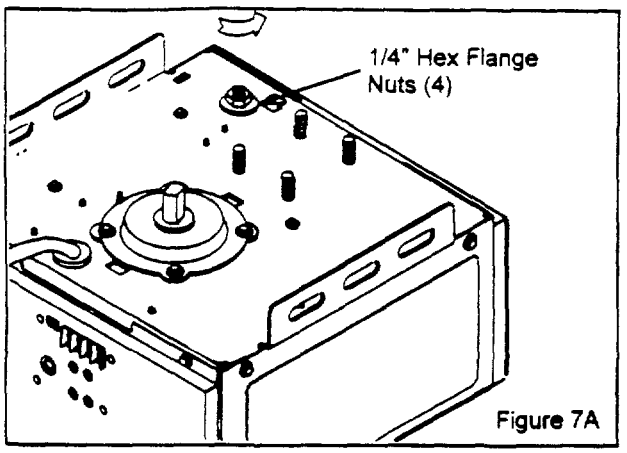


Figure 7A

Step 1.
Place Power Head Unit on discarded packing material/ cardboard. Remove the four (4) 1/4" Hex nuts See Figure 7A. Save for use later.

Step 2.
Remove L-Rail/Chain Speed Unit from box Locate Installation Hardware Package.

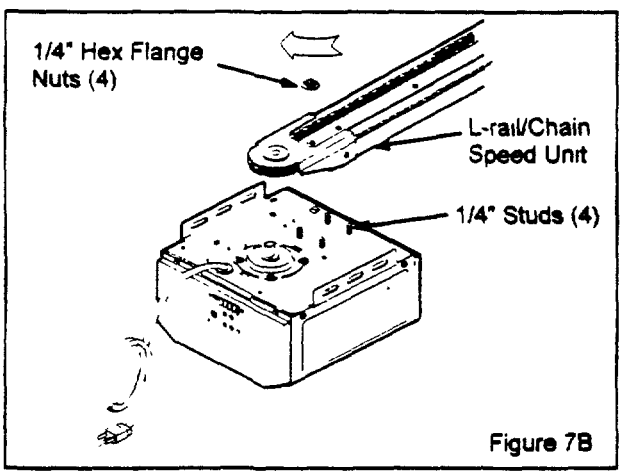


Figure 7B

Step 3.
Since L-rail/Chain Speed Unit comes pre-assembled from factory, simply position L Rail / Chain on top of the four (4) 1/4" studs on the Power Head Unit Chassis (See figure 7B), Secure with nuts and washer removed in Step 1

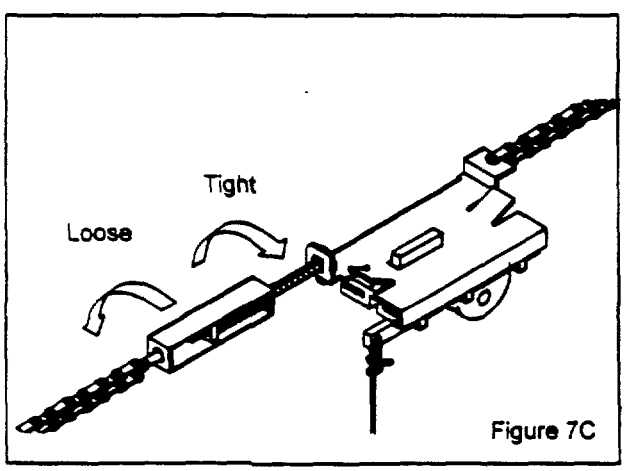


Figure 7C

Step 4.
Check chain tension. When adjusted properly, the chain should show no drop and be approximately 1/2" above the base of L-Rails. If necessary, slowly apply or release tension to the chain by turning the Chain Tension Bracket (See Figure 7C). Make sure the chain does not twist and re-check chain alignment at drive sprocket and front idler Note: Improper chain adjustment (too loose or too tight) can result in improper operation and/or excessive sprocket and chain noise.

Step 5.
Re-check all nuts for tightness.

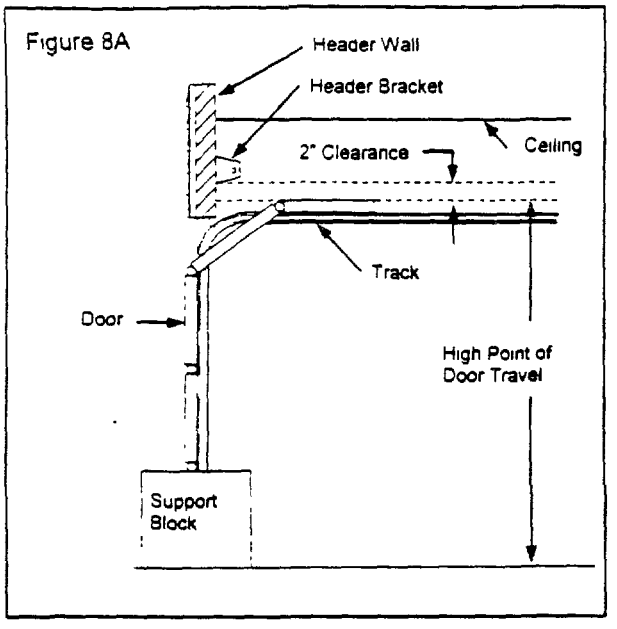
Assembly is now completed. You are ready to begin installation of the opener.

Installation

Installation procedures will vary depending on type of garage door
Identify your garage door from those illustrated below and follow procedure outlined for your type of door

To determine high point of door Raise the door slowly, until it reaches its highest point of travel Place support block under the door and measure the distance from the floor to the top edge of the door Remove block and lower door

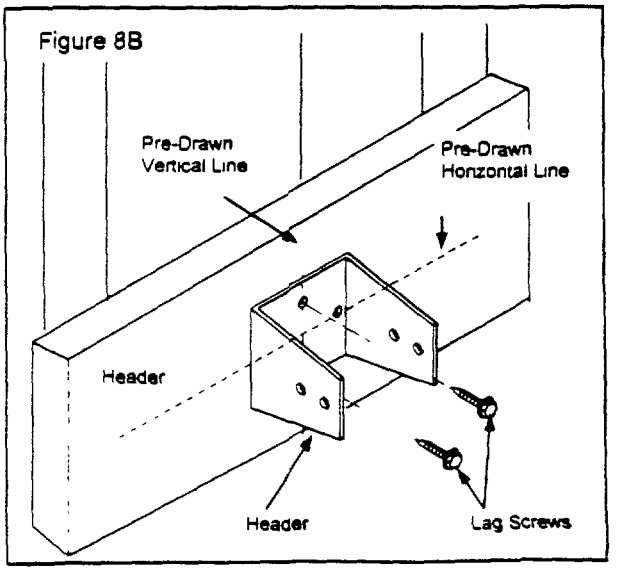
Figure 8A



Sectional Door with Curved Track

Springs, Pulleys, Cables and Mounting Hardware that balance your garage door are under tremendous pressure at all times and can cause serious injury or death if disturbed. Do not attempt adjustment!

Figure 8B



Step 1.
Mounting the front Bracket
Mark a vertical centerline on the header above the door. By manually raising the door, determine the high point of the doors travel (see figure 8A) and using a level, transfer this measurement to the header (See figure 8B) Draw a horizontal line, crossing the previously drawn centerline, at this point. Install the front Mounting Bracket securely with lag screws as shown below. If necessary, reinforce the header with steel or wood to ensure a secure mount.

LX 1097

Installation

Warning: Fiberglass, Aluminum or Lightweight Steel Doors Will Require Reinforcement before installation of door mounting brackets. Contact your door manufacturer for a reinforcement kit or instructions. Failure to properly do so may result in severe door damage.

Note: Reinforcing may affect the balance of your door. Check for proper manual operation after installation. If necessary have your door re-balanced by a qualified garage door service person.

Figure 9A

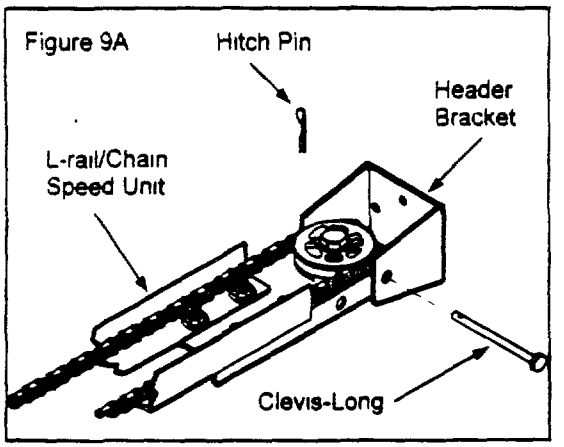


Figure 9B

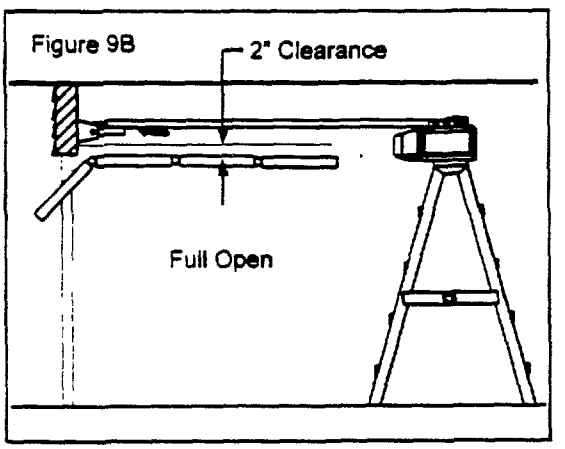
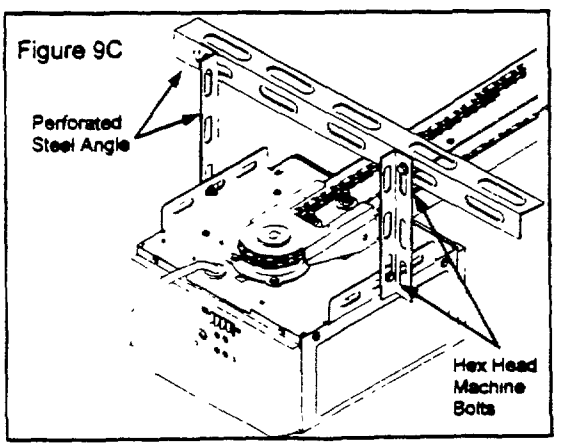


Figure 9C



Step 2.

Place the power unit on the floor (use cardboard packing material for protection) and raise Mounting Bracket align. Insert 5/16 x 2-3/4" Clevis Pin and Hitch Pin. See Figure 9A.

Step 3.

Raise the opener and rest the power unit on a ladder or other support. Open the door to the full open position. Allow 2" of space between the rail and the door. See Figure 9B

Note: Since the opener will be secured permanently in this position, open and close the door a few times to be sure the door does not rub on the rail and that you have allowed the proper clearance before proceeding.

Step 4.

Mount Power Head to Ceiling:

Since there are many different ceiling designs, all possible mounting illustrations can not be shown. Primary concern is to secure the power head to the ceiling so that operational strength, rigidity and safety are achieved. The opener must be securely fastened to a structural support of the garage. Although there are a series of mounting slots provided on the power unit, try to secure the mounting hangers to the slots closest to the front. Mounting may usually be accomplished by using standard 1-1/4" perforated steel angle available at most hardware stores or from your local garage door service person. If in doubt as to location of, and attachment to, ceiling joists, a carpenter should be contacted to provide assistance. A cross support will be necessary if power head is mounted 8" or more from the ceiling. See figure 9C.

Fasteners Supplied: 2 ea. 5/16-18x 1-7/8" Lags, 4 ea. 5/16 - 18 x 1-1/8" Hex Bolts with Lock Washers and Nuts.

Step 5.

Re-check all Lags, Nuts and Bolts for Tightness.

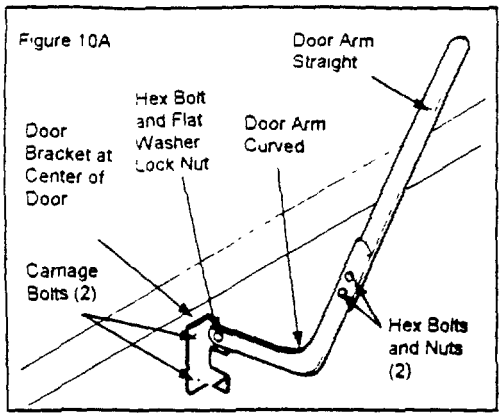
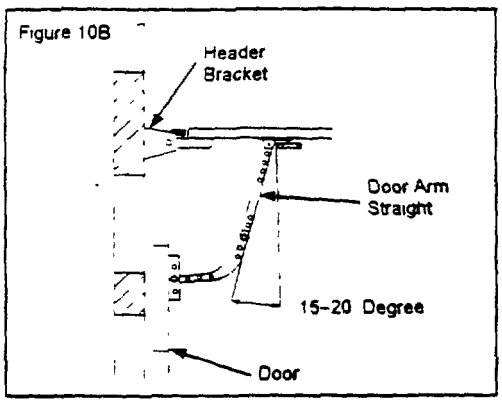


Figure 10A
Door Arm Straight
Hex Bolt and Flat Washer Lock Nut
Door Arm Curved
Door Bracket at Center of Door
Carriage Bolts (2)
Hex Bolts and Nuts (2)

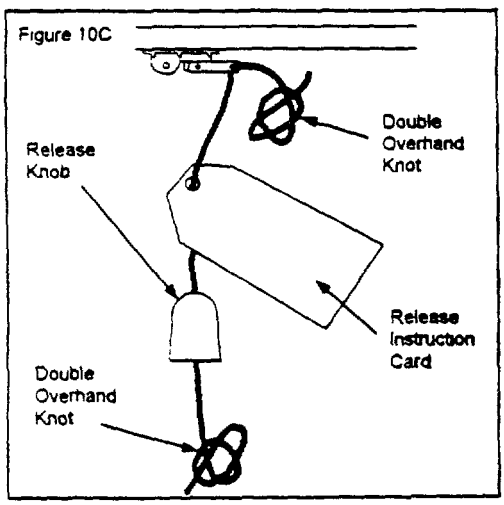
Step 6.

Door Arm, Bracket and Plate Installation:
Install the door mounting bracket on center and even with the top set of rollers on the door as illustrated Figure 10A (Fasteners Supplied. 2 ea Carriage Bolts and Nuts 1/4-18 x 2-1/4" with Flat and Lock Washers) Connect the straight door arm section (single hole section) to the trolley using a 5/16 x 1" Clevis pin and clip The door arm must pivot freely Connect the two Door arm so that it leans toward the power unit as illustrated fig. 10B Do not install the door arm so that is straight up and down when the door is closed or the emergency release will not function properly Now attach Door Arm to Trolley using 5/16" x 1" Clevis pin and clip



Step 7.

Tie a double overhand knot in one end of the Emergency Release Rope and slip the other end through the Red Release Knob, the Release Instruction Card and the hole at the end of the Release Lever on the Trolley (Please take time now to read and familiarize yourself with the instructions on the Emergency Release Label) Figure 10C Tie a second double overhand knot in the free end, adjusting the Rope so that the Red Knob is 6 FT above the floor If the Rope must be cut, flame seal the cut end with a match or lighter to prevent fraying or unraveling



Note: The Emergency Release Mechanism is engaged by pulling the Release Knob down and towards the door, allowing the Trolley Mechanism to separate freeing the door To re-engage simply move the Emergency Release Mechanism Lever upwards and operate the Opener using the push button or Transmitter The two parts will automatically reconnect.

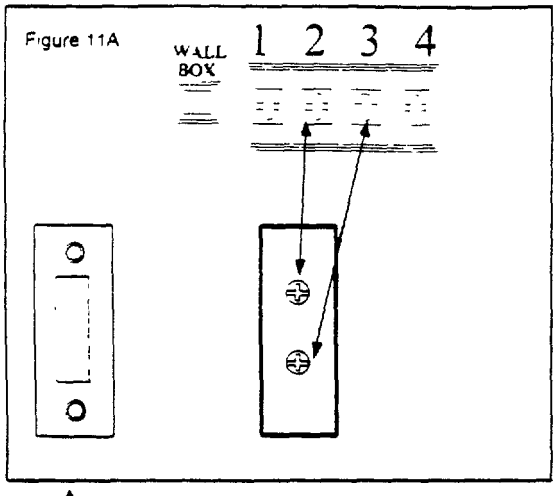
Note: This operation should be attempted only when the door is fully closed. Disconnection when open or partially open can cause the door to close quickly and cause personal injury or damage to the door.

Use Manual Release Rope to Disengage Trolley Only. Do Not Use the Rope and Knob to Pull the Door Open or Closed

Warning: A Child Operating the Door Controls Risks Injury Or Death to Himself and to Others Do not Allow Children to Operate Any Door Controls Mount the Push Button at Least 5 FT from the Floor Out of Reach of Children

Warning: Improper Door Operation can Cause Injury or Death Caution Label Must Be Mounted on Wall Near The Push Button. All Warnings Should Be Strictly Adhered To

Figure 11A



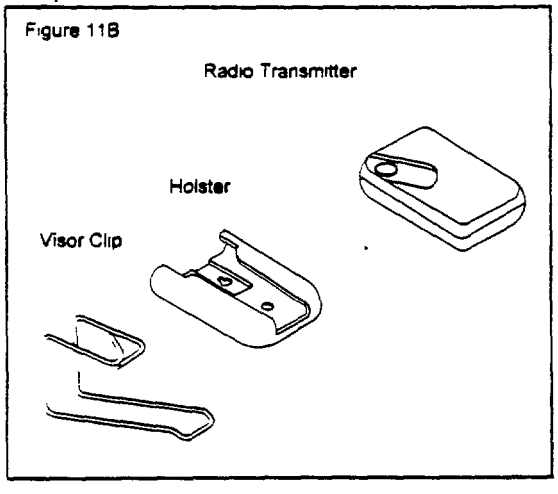
Step 8:

Installing the Standard Wall Push Button

Remove about 1/4" of insulation from both ends of the 2 strands low voltage bell wire Connect one end to both screw terminals on the back of push button Select a convenient place near an access door 5 ft from the floor and out of reach of children for mounting. (2) 1/1/2" Screws supplied Install the Caution Label Figure 14B page 14, near this installation Run bell wire up the wall and across to the opener Secure with insulated staples Attach to Terminals 2 & 3 Figure 11A

Warning: Installation & Wiring Must Be in Compliance with Local Electrical and Building Codes Operation at other than 120V 60Hz Will Cause Opener Malfunction and Damage.

Figure 11B



Step. 9

Opener Must be permanently wired or plugged into a grounded 3-prong receptacle wire according to local codes, within 3 Ft of the Power Unit. A GFI Type Receptacle 14 Ga. or heavier is Recommended. Do Not Use a 2-prong adapter or Extension Cord. If local Code requires permanent wiring, a GFI Type is recommended to protect the line Contact a Licensed Electrician to install required circuit and to direct wire the Opener

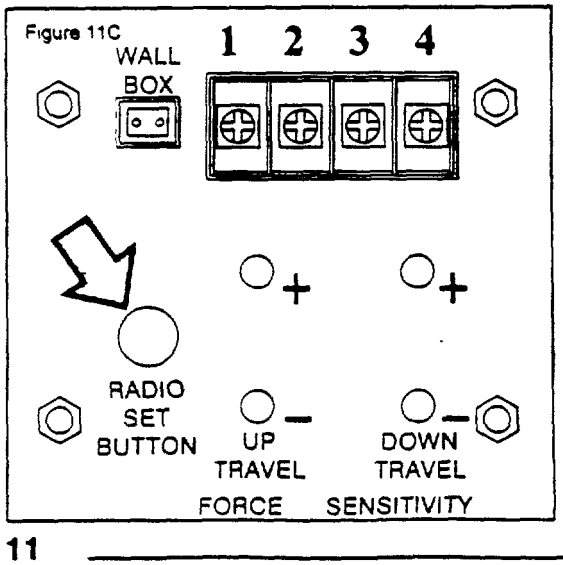
Step. 10

Install a 60 watt Rough Service Bulb (available at most hardware stores) firmly in the light socket. Light bulbs in Door Openers are subject to vibration during normal operation which may shorten their life span. Fit Light Lens Cover and snap into place.

Step 11

Setting Personal Transmitter Access Codes.

Figure 11C



Your Opener contains a built in Receiver

To set Receiver:

With a pointed object push the button shown in figure 11C If the courtesy light is off it will come on and stay on for 30 seconds. If the courtesy light is on it will flash once and then stay on for 30 seconds. During this 30 second time period push the round button on a transmitter (figure 11B) until the courtesy light flashes twice this will indicate that the receiver has accepted the code of the transmitter If the code is not accepted the courtesy light will stay on for 30 seconds, flash 4 times and then stay on for 4 1/2 minutes. Up to 6 transmitters (including wireless keypad) can be programmed into the receiver in this way by repeating the process Each transmitter has its own code set at the factory. If you enter more than 6 transmitters the oldest transmitter will be dropped from memory and the new transmitter will be added. To delete all transmitters from memory hold the receiver set button in for 5 seconds The courtesy light will then flash 7 times, indicating all transmitters have been cleared from memory.

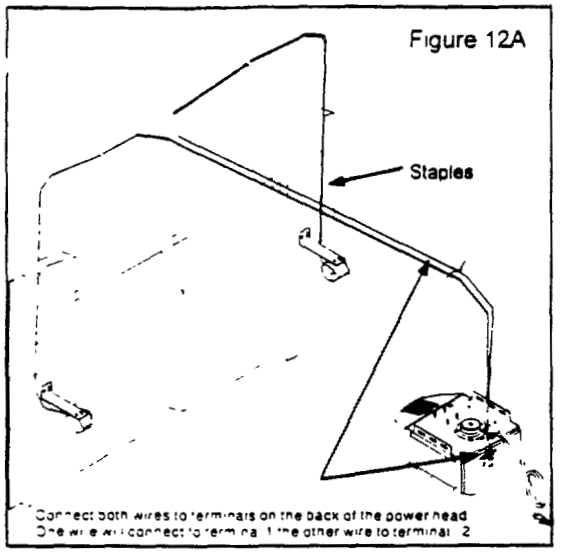
Installation

Your Garage Door opener is supplied with an Auxiliary Entrapment System Please Read The Following Carefully

Important: The Opener will not operate until the Silent Guard Photoeyesystem is installed properly aligned and connected

The Beam must not be obstructed in any way If obstructed the door can only be closed by applying constant pressure to the wall button only The Transmitter cannot be used to close the door

Warning: A Garage Door without an Auxiliary Entrapment Protection System Poses a Threat of Injury or Even Death Install the Photoeyesystem No Higher than 4"-6" Above the Floor To Reduce All Risk to Children
Risk of Entrapment. Disconnect Power to the Opener Before and During Installation of This Accessory Do Not Reconnect Power to Opener Until Instructed To Do So Ensure Doorway is Clear Before Starting Testing of Unit.


Staples
Connect both wires to terminals on the back of the power head
One wire will connect to terminal 1 the other wire to terminal 2
Figure 12A

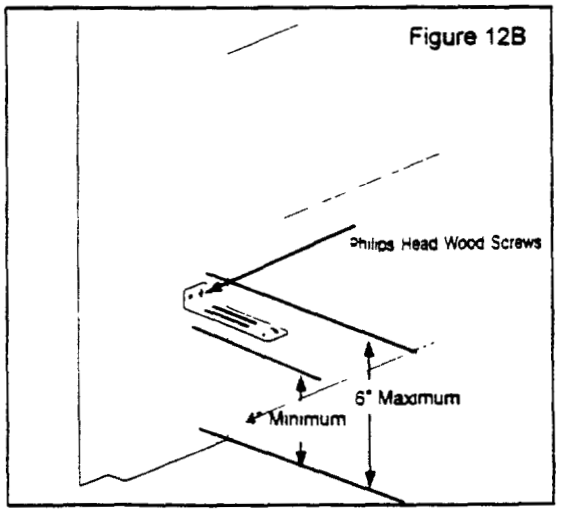


Figure 12B

Step 12

Installation of Silent Guard Photoeyesystem

A. Mark the position of the Silent Guard Photoeyesystem as follows Mark a line on the left and right door jamb (as close as possible to the door track) 4" and 6" above the floor (See Figure 12B). The top mark is the maximum height and the bottom the minimum height that the Silent Guard Photoeyesystem may be placed

B. Mounting the Silent Guard Photo Eye System L Brackets:
1. Remove the two-L mounting brackets from the carton as shown in figure 12B.
2. Attach L mounting brackets to the jamb with Philips head wood screws supplied as shown in figure 12B 4" to 6" above the floor.Make sure that the brackets are the same distance from the floor on both sides of the doorjamb and straight.

C. Connecting the Silent Guard Photo Eye System
1. Disconnect bell wire from the terminals on the back of the photo eyes.
2. Running wire for connections to power head and photo eyes: For your convenience wire connections are none-polarity sensitive universal wiring.With the bell wire supplied start at the power head connecting one wire to terminal number 1 the other to terminal number 2 as marked on the power head as shown in figure 13B on page 13.
3. Using the 3 wire clip brackets supplied attach one side of the clip to the operator rail assembly in equal distances between the power head and header bracket.
4. Insert the bell wire into the other side of the wire clip and run the length of the operator rail assembly to the garage door header. Using the insolated staples supplied run the bell wire across the header and down the wall to one of the photo eye mounting brackets. Cut bell wire making sure there is enough to reach to the end of the mounting bracket.
5. Strip both ends of the bell wire 1/4" and attach both sets of wires to terminals on the back of the photo eye. There will be two wires connect to each terminal, one strand from each pair of wires.
6. Insert screws coming out of the top of photo eye into the holes in the mounting bracket and twist the photo eye to engage into the bracket as shown in figure 13A on page 13 through the holes in the bracket and twist.

Figure 13A

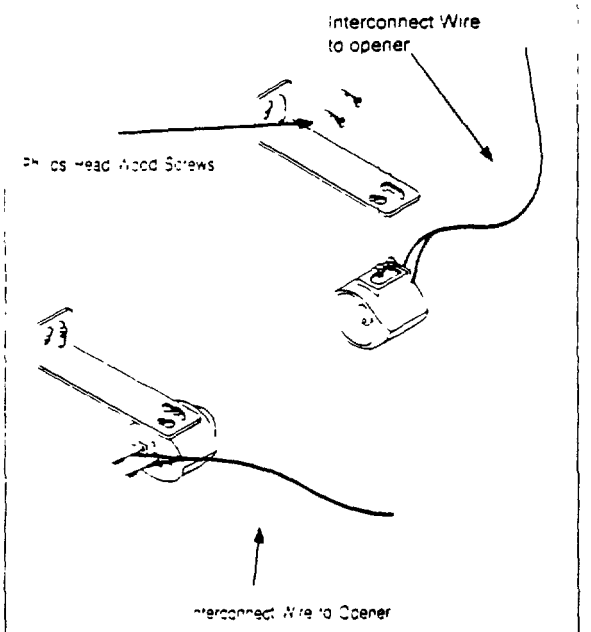
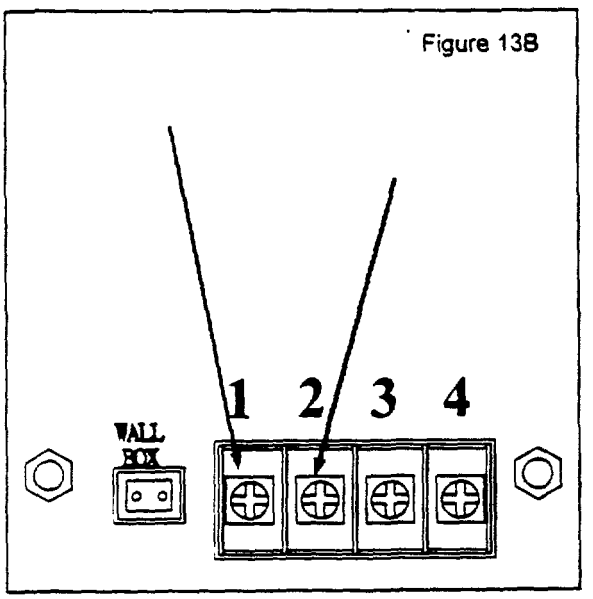


Figure 13B



7 Run the remaining bell wire up the wall and across to the other photo eye mounting bracket using the supplied insolated staples

8 Strip 1 4" of the insulation off the wire and attach to terminals on the photo eye Repeat photo eye mounting as before

Note. This is the preferred method of wiring Alternative wiring methods are optional. The photo eyes system will work properly as long as there are wires to all three components in a loop that make up the photo eye safety system Terminals 1 and 2 on the back of the power head and terminals on the back of both photo eyes

D. Final Alignment and Test

1 Connect power to the opener The light on the opener will flash for 5 seconds indicating power has been restored to the unit. Keep a transmitter with you to control the opener. Now that you have power to the photo eyes the LED close to the terminals on the eyes will be lit up If the LED doesn't light up check terminal connections and wire.

2. Go to the receiver side of the photo eyes (eye with LED in the front).Aim towards the other eye as close as possible and tighten down the screws holding the eye to the mounting bracket.

3. Go to the transmitter side of the photo eyes (other side) and aim the eye by twisting slowly and aligning with the receiver When the LED in the receiver comes on you are aligned. Tighten down screws on the transmitter

4. Place your hand or a solid object one foot in front of the transmitter or receiver. The red LED should go OFF and remain OFF until the object is removed NOTE. There may be a slight delay in returning to normal depending upon how long the photo eye system was blocked.

5. Move to the center of the door. Make sure the red LED light is On. Move your hand or a solid object slowly through the beam. The LED should go Off and On If not, check alignment of the system.

6. At this time set the down limit Page 14 and reversing system adjustment Page 15 the opener It is very important that the Openers inherent features operate as intended before completing the photo eye system tests

7. Place an object at least 6" high on the floor at the center of the door. Now, attempt to close the door The door should not close from the portable transmitter, but will close with constant pressure from the wall button

8 Remove the obstruction from the photo eye system beams path. Close the door. Towards the bottom of the doors downward travel, Carefully move your hand across the path of the beam at the center of the door The Door should stop, pause for approximately 1-1/2 seconds and Reopen. Retest, breaking the beam one foot in front of both the transmitter and receiver unit while the door is moving downward. The door MUST STOP and OPEN each time. If not, re-align the photo eye system until proper operation is obtained.

9. Tighten all mounting screws, nuts and bolts

Adjustment

Warning: If Limits are not adjusted properly, the emergency release mechanism may not work properly and door operation could result in door damage or serious injury

Note: It is now necessary to turn on the power in order to run the opener to check limit settings Before doing so ensure that all mounting hardware is installed and has been properly tightened, that all electrical connections are per local code requirements. Double Check that all ropes have been removed and that the doorway is clear

Figure 14A

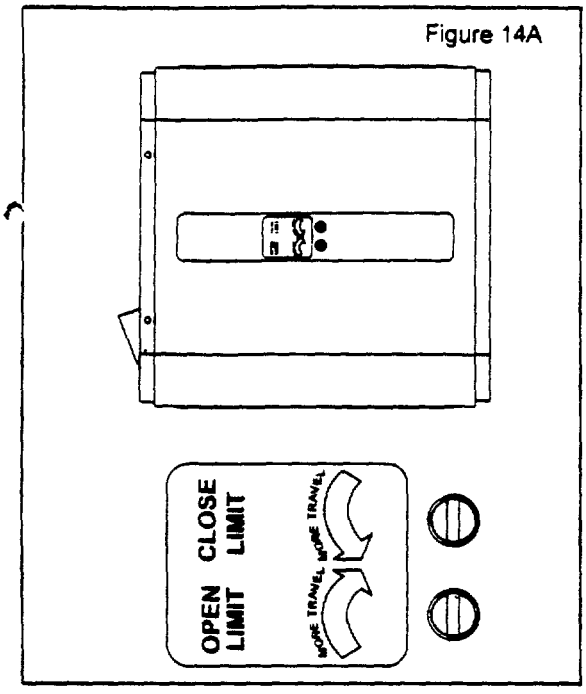
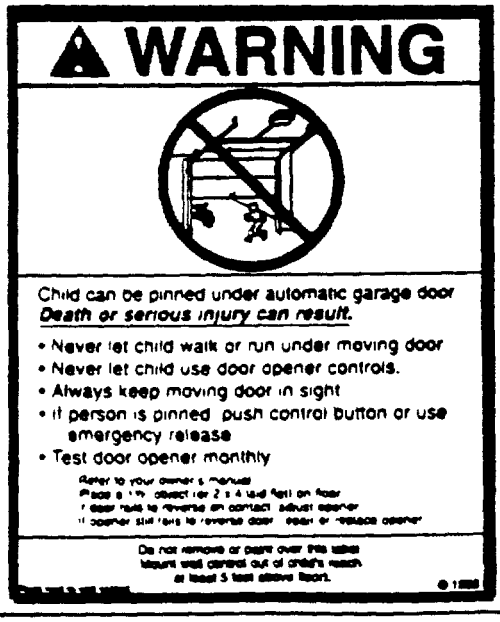


Figure 14B

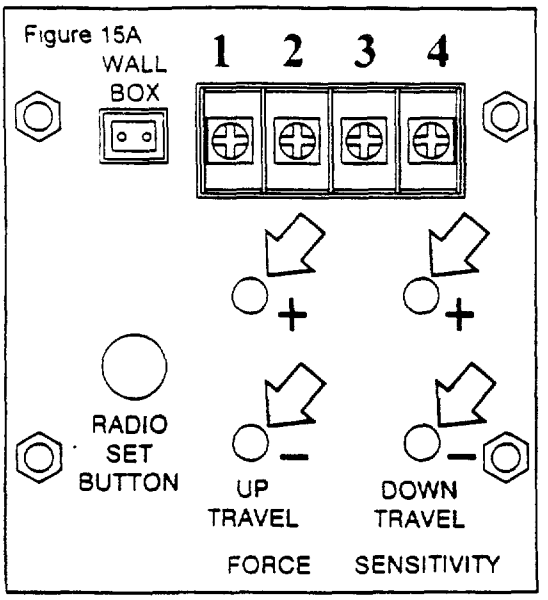

WARNING
Child can be pinned under automatic garage door
Death or serious injury can result.
• Never let child walk or run under moving door
• Never let child use door opener controls.
• Always keep moving door in sight
• If person is pinned push control button or use emergency release
• Test door opener monthly

- Up (Open) and Down (Close) Limits
- Limit adjustments settings regulate the total amount of distance the door will travel while opening and/or closing.
- Setting the Up Down Limits
- Note: Run the Opener through a complete travel cycle after each adjustment.
- Note: Repeated Operation of Opener may cause the motor to Overheat and the Thermal Overload Protection System will shut off the Opener. Please wait approximately 15 minutes for the Opener to cool down and automatically reset.
- If the door opens completely and closes completely without reversing: No adjustment is necessary.
- If the door does not open completely but opens at least five feet: Increase Up travel by turning UP LIMIT screw Clockwise
- If the door does not open five feet: Adjust Up FORCE/ SENSITIVITY Setting Page 15.
- If Door does not close completely: Increase Down Travel by turning DOWN LIMIT screw Counter-Clockwise.
- If Door reverses in full closed position: Decrease Down travel by turning Down LIMIT screw Clockwise.
- If Door Continues to Reverse: Disconnect Door from Opener use manual release. Check for binding while working door manually. If door binds or is out of balance call a qualified garage door service person. If no binding is found and balance seems correct. Adjust FORCE/SENSITIVITY Setting Page 15.

Force/Sensitivity Adjustment

Warning: Improper Adjustment of Sensitivity System Force Could Cause Entrapment, Injury or Death Set Adjust For Just Enough Force To Operate The Door Reliably, But No Stronger Do Not Over-Adjust Force/Sensitivity System To Compensate For A Poorly Working, Sticking or Binding Door (Contact a qualified Garage Door Service Person to Correct any binding, sticking and/or other door problems)

Figure 15A



Force/Sensitivity Adjustment

- Force/Sensitivity Adjustment Controls are located on the Back Panel of the Opener (See Figure 15A) Up and Down sensitivity adjustment are independent of one another (+ indicates more force) and (– indicates more sensitivity) Never change the Force/Sensitivity setting by more than 1 increment at a time.
- **Note:** Force/Sensitivity is set at level 1 at the factory both for the up and down adjustments The size and operating conditions of the garage door will determine the correct level of Force/Sensitivity. Weather conditions may also affect the operation of the door requiring further adjustments as needed.
- **Note:** If the down pressure adjustment is set too light the operator will reverse in the down cycle. If the up pressure adjustment is set too light the operator will stop in the up cycle
- **Note:** There are 12 levels (increments) of up and down force/sensitivity setting. When either the maximum or minimum setting is reached the light will flash 6 times.

Setting Force/Sensitivity Adjustment

Down force/Sensitivity Adjustment

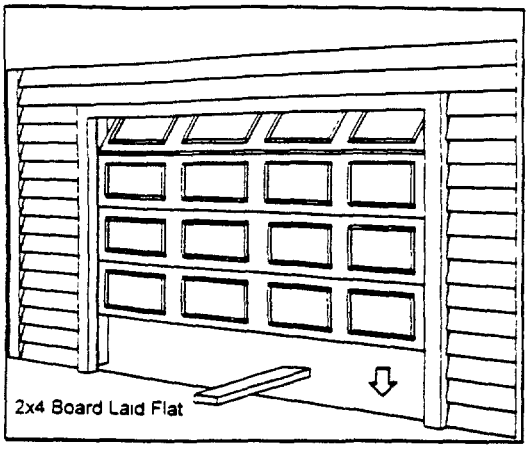
- In the down or close direction the operator will reverse if the force required to close is exceeded. If the operator completes a full closure without reversing decrease the amount of pressure in increments of one push of the down — button until the operator reverses. Once the operator does reverse in the down direction increase down pressure by pushing the down + button once.

Up Force/Sensitivity Adjustment

- In the up or open direction the operator will stop if the force required to open the door is exceeded If the operator completes a full open cycle without stopping decrease the amount of force in increments of one push of the up – button until the operator stops during the up cycle. Once the operator does stop increase the up force by pushing the up + button once. For safety the force setting should always be set at the least amount necessary to run the door properly.

Testing the Safety Reverse System

Warning: Failure to Test Reversing System Could Result in Death or Serious Injury Test This System Once A Month

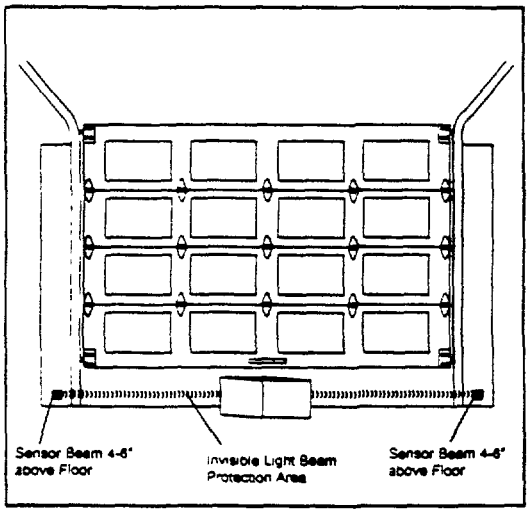


To test the Opener Reversing Feature at floor level place a solid object 1-1/2" thick on the ground so that the center of the door will contact it. Close the Door If the Down force Adjustments are Correct, the door will reverse within one and a half (1-1/2) seconds of contacting the object and travel to the Full Open Position Light will also start flashing and continue to flash for 60 seconds If this does not occur, re-check Limit Adjustments Page 14 and Force/Sensitivity Adjustments Page 15.

Note: Any time any adjustments are made to Limits or Sensitivity, You Must Retest the opener for the Reversing Feature at Floor Level as outlined above.

Warning: The Sensitivity System Reversing Test Should Be Performed Monthly to Ensure That This Important System Remains in Proper Adjustment

Warning: A Damaged or Malfunctioning Photoeyesystem Could Enable a Garage Door to Close on People or Property, Causing Serious Injury or Death Perform this Test Monthly to Ensure Proper Operation.



Testing the Photoeyesystem

Start the door down and then place an obstacle approximately 8" high by 12" wide in the path of the beam The Red Pilot Light on the Silent Guard Photoeyesystem should go off The door should stop for 1-1/2 seconds and reverse to the full open position. The Opener light will also begin flashing and continue to flash for 60 seconds. IF the door is moving up and the beam is broken, the door will continue up to full open With the door fully open and at rest, place the obstacle in the path of the beam once again. Activate the Wall Push Button, The Opener will revert to and remain in the safety. Push And Hold Operation for close travel. See Page 13.

NOTE: If the Garage Door Travels More Than One Inch in a Downward Path After Releasing the Button, the Silent Guard Photoeyesystem is Malfunctioning Check All Electrical connections and Alignment of the Photoeyesystem.

Final Installation

Make Sure That:

1. The front and rear mounts for the opener are sound and secure and the rail is positioned correctly above the high arc of the door and that the opener is positioned over the door action centerline

2. For sectional doors , the position of the door arm with the opener closed, is such that its connecting point on the trolley is 5" to 8" behind it connecting point on the door bracket. The door arm should never be perfectly vertical when the door is in the closed position.

3 The emergency release knob and cord are secured to the emergency release lever on the trolley The knob is located 6 ft above the floor and requires no more than 50 lbs. pull to actuate. The trolley and release mechanism are properly lubricated.

4. The standard lighted wall push button is in such a position and at such a height that it can only be actuated by an adult. The caution label is prominently displayed next to the push button.

5. All wiring is correct to code. There is ground continuity in the supply. The ground prong on the power cord is intact.

6 All ropes have been removed from the door. The door moves freely without binding when operated manually The door is correctly balanced and lubricated. All door hardware is secure and sound. The sensitivity has been adjusted to minimum force. The appropriate warning sticker has been affixed to the door.

7 The door reverses on obstructions to within 1" of the floor. The floor beneath the closed door provides uniform contact.

8. The plastic envelope with this manual is attached to the wall near the push button.

9 On doors with extension type springs, safety restraint cables have been installed through the springs.

10. There is GFI protection on the power line to the opener.

Trouble Shooting

Symptoms	Probable Cause/Solution
Opener does not work from either the transmitter or push button	1 2.3.5.6.8 11.17 23 24
Opener does not start on command and light flashes for 4 seconds	1 4.5.7 8 11 12 13.15 16 17 18 22 23 24
Opener stops during cycle and light flashes 4 times	6
Opener operates from push button but not radio	9.14 20 21 25
Stops before reaching full open or closed	10 15
Reverses when the door makes contact with the floor	10
Light flashes 5 times but the opener does not start	26
Light does not come on	5.19
Light will not turn off (Light stays on)	5

Probable Cause	Solution
1 Mechanical garage door lock is engaged	1 Remove all locks from the garage door
2 120 volt power is not present at outlet.	2. Check circuit breakers, fuse box and GFI receptacle
3 Shorted or defective push button.	3. Remove push button from wall and disconnect wires from the push button (activate transmitter) If the operator works from transmitter replace push button
4 Bell-wire is shorted	4 Disconnect bell wire from the terminal stripe on the back of the power head (activate transmitter).
5. Malfunctioning logic board	5 Unplug operator at 120-volt outlet and then reconnect Consult repairman to replace logic board.
6. Thermal overload breaker protection has been activated.	6. Allow time for motor to cool down (15 to 30 minutes depending on temperature in the garage) Thermal breaker automatically resets itself.
7 Garage door springs are out of balance	7. Consult service repairman to balance or replace springs Operate garage door manually; disconnect opener by pulling red cord on the trolley. Look for binding or jamming of the garage door. Consult service repairman
8 Garage door is jammed.	8. . Replace 12-volt battery in the transmitter by removing the screws on the backside of the transmitter case and pulling the case open to expose battery.
9. Weak transmitter battery	9. Refer to page? In the installation manual.
10. Travel limits are out of adjustment	10. Check for obstructions between the photo eyes and remove obstruction.
11 Photo eyes are obstructed (light flashes 30 seconds)	11 Check red LED inside the lens of the receiver photo eye (photo eye with 4 wires). Led will light up when photo eyes are aligned properly, refer to page 12 & 13 In the installation manual.
12. Photo eyes are out of alignment (light flashes 30 seconds)	12. Replace existing bell wire with new. Make sure that insulated staples do not short wire.
13. Photo eye bell wire is shorted (light flashes 30 seconds)	13. Contact your local dealer.
14 Defective transmitter or receiver	14. Adjust up force/sensitivity see page?
15 Up force/sensitivity out of adjustment	15. Adjust down force/sensitivity see page?
16. Down force/sensitivity out of adjustment	16 Free bottom of the door from the ice.
17 Bottom of garage door frozen to the ground	17 Clear all foreign objects from under the door
18 Foreign object on the floor such as ice or snow.	18. Replace with rough service bulb of 75 watts or less.
19 Defective light bulb	19. Point wire antenna straight down towards the floor
20 Radio receiver is not receiving the radio signal	20. Ensure radio signal (transmitter) has a direct line of site to the receiver. Radio signals do not pass through metal objects but do penetrate glass.
21 Location of radio transmitter	21. Contact your local dealer.
22. Defective RPM counter board	22. Contact your local dealer.
23 Defective programming board	23. Contact your local dealer.
24 Defective photo eyes	24 Contact your local dealer.
25. Defective transmitter	25. Contact your local dealer.
26 Vacation switch is activated on the deluxe 3 function wall stations.	26. Switch back to Vacation off position.

After the Installation

1 Never permit children to play with or operate the garage door opener either from the wall station or the remote controls Keep radio transmitter locked in the car Misuse of the push button or transmitter could result in serious injury or even death

2 Open the garage door with the remote controls only when the garage door is fully visible to you and clear of all obstructions The garage door should be kept in sight until it is completely open or closed and you are certain that the garage door opener has shut off

3 Attempting to exit the garage through the garage door opening, while the door is in motion, is a very dangerous activity that could result in serious injury or even death.

4 Children and pets should always be clear of the door opening while the garage door is in motion.

5. Check the safety reverse mechanism at least once a month to make sure that it will reverse with the minimum amount of force. Also check to be sure that the door will reverse within 1-1/2" of the floor. See page 20.

6. Check the manual operation of your garage door at least every 90 days to be sure that it is operating smoothly and does not bind or stick. Tighten all bolts on the door and visually check all hardware including springs for wear of damage. Caution: If service is needed contact your local garage door service person

7. Do not decrease the safety reversing sensitivity mechanism to overcome a damaged or poorly operating door This will adversely affect the operation of the safety reverse mechanism which could result in damage to the door, personal injury or even death.

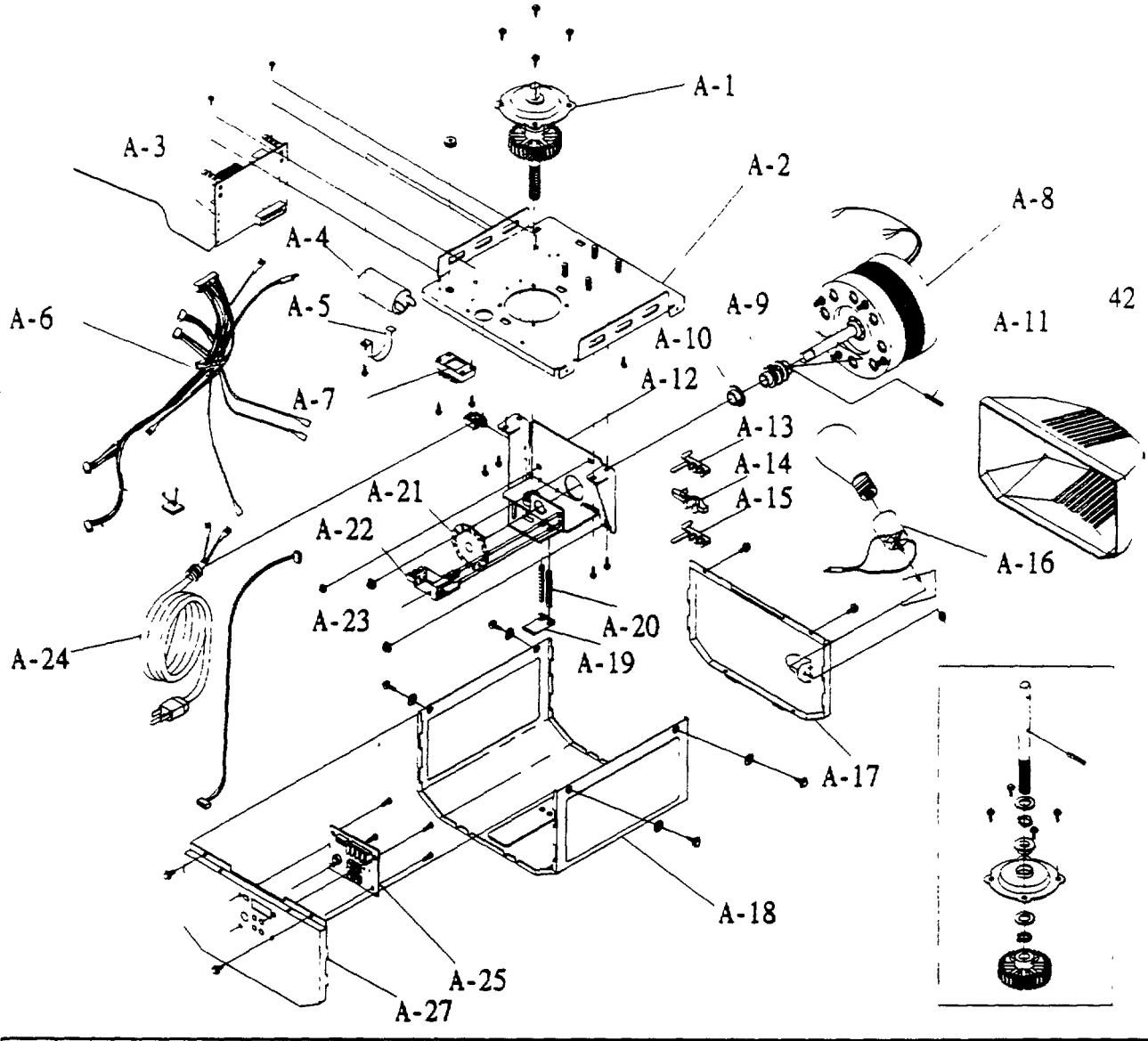
Caution: Never operate the door opener if the reversing mechanism is not functioning properly.

8. Whenever possible, the manual disconnect should only be used when the door is fully closed. Caution: Extreme care must be taken whenever the disconnect cord is pulled with the door partially open. Weak or broken springs may allow the door to fall rapidly resulting in property damage, personal injury or death. If a broken spring is evident, contact your local garage door service person immediately before disconnecting the door from the opener. Never attempt servicing a broken spring.

9 Always disconnect electrical power supply to the opener when performing any maintenance or service to the opener or garage door. Failure to do so could result in electrical shock, property damage, personal injury or death

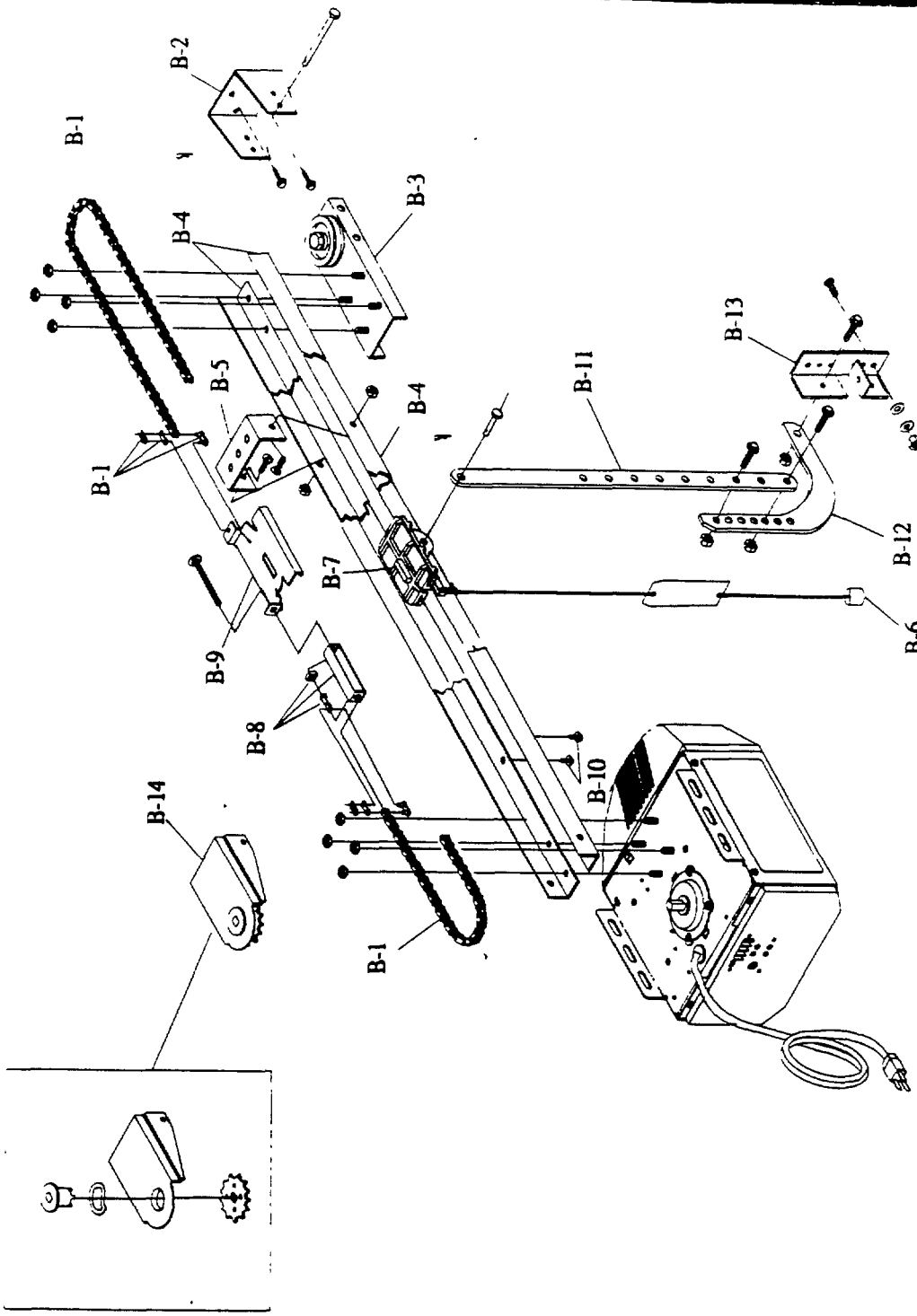
10. If any damage to any mechanical or structural component of the opener is observed, discontinue use and contact your local garage door service person immediately.

Rail and Hardware Parts List



A-1	Drive Shaft Assembly w/ Helical Gear	A-10	Motor Shaft Bushing	A-19	Limit Shaft Clip
A-2	Chassis	A-11	Roll Pin	A-20	Limit Screws
A-3	Logic Board	A-12	Motor Mounting Bracket	A-21	RPM Reader Wheel
A-4	Capacitor	A-13	Up Limit	A-22	RPM Reader Board
A-5	Capacitor Mounting Bracket	A-14	Limit Actuator (Traveler)	A-23	RPM Reader Board Bracket
A-6	Wiring Harness	A-15	Down Limit	A24	Power Cord
A-7	Thermal Block	A-16	Light Socket	A-25	Programming Board
A-8	Motor	A-17	Front Panel	A-26	Back Panel
A-9	Worm Gear	A-18	Wrap Cover Assembly	A-27	Light Lens

Rail and Hardware Parts List



B-1	Chain	B-6	Emergency Pull-Cord with Knob	B-11	Straight Door Arm
B-2	Header Bracket	B-7	Trolley Slide Assembly	B-12	Curved Door Arm
B-3	Pulley Assembly	B-8	Chain Tension Assembly	B-13	Door Bracket
B-4	"L" Rail	B-9	Inner Trolley (Metal)	B-14	Drive Sprocket Assembly
B-5	Rail Spreader	B-10	Trolley Stop		

Limit Warranty
Residential Garage Door Opener and Remote System

Lynx Industries warrants all components of its residential garage door openers against defects in material and/or workmanship to the original purchaser Any parts, power unit, remote control system or complete garage door opener that fails during the warranty period will be repaired or replaced in accordance with the following provisions

LPL 50 Deluxe 1/2 Horsepower
Motor· Warranted to be free from any defect in material and/or workmanship for a period of 5 years from date of purchase
Drive Train· ALL Mechanical parts to be free from any defect in materials and/or workmanship for a period of 3 year from date or purchase.
Electronics All electronics parts to be free from any defect in material and/or workmanship for a period of 1 year from date of purchase

LPL 33 Standard 1/3 Horsepower
Motor· Warranted to be free from any defect in material and/or workmanship for a period of 3 years from date of purchase.
Drive Train ALL Mechanical parts to be free from any defect in materials and/or workmanship for a period of 1 year from date or purchase.
Electronics All electronics parts to be free from any defect in material and/or workmanship for a period of 1 year from date of purchase

If within the stated warranty period it is determined that a component is defective or that a component failed as a result of poor workmanship **Lynx Industries** will repair or replace (**Lynx Industries'** option) the component at no cost to the owner. Defective parts will be repaired or replaced with new or factory rebuilt parts.

To make claim under this warranty contact **Lynx Industries** you will be instructed on how to proceed regarding replacement and repair.

This warranty is void when any component part fails because of damage from improper use, handling, faulty installation, faulty connection to an improper power source, failure to properly maintain, modification of unit, use other than on residential garage door and damage from fire, water or acts of God.

This warranty excludes all other warranties, expressed or implied, including any warranty of merchantability **Lynx Industries** shall not be liable for consequential or incidental damages that may arise from any defect or failure of the garage door opener. No person or representative is authorized to extend, modify or change the provisions of this warranty in connection with the sale installation or use of this product.

All labor charges for removal and re-installation of the door opener are the responsibility of the owner **Lynx Industries** shall not be responsible for the cost of any part, labor or service charge which is performed by any other person or entity at any time.

Batteries are not included in the warranty program.

Some states do not allow the exclusion or limitation or consequential or incidental damages. This Limitation is not valid in jurisdictions which do not allow limitation of incidental or consequential damages or limitation or warranty periods.

This warranty give you specific legal rights and you may also have other rights which vary from state to state.

檔名:	DOOR
目錄:	C:\My Documents\SourceCode
範本:	C:\Program Files\Microsoft Office\Templates\Normal.dot
標題:	
主旨：	
作者:	
關鍵字:	
註解:	
建檔日期:	00/04/17 01:08 PM
修訂版編號:	1
前次更新日期:	
前次存檔人員:	
編修總時間:	0 分鐘
最後列印在:	00/04/17 01:10 PM
最後列印的字數	
頁數:	7
字數:	837 (約)
字元數:	4,775 (約)

CONFIDENTIAL
LX 1112

```
$BASE 10T
,   door program
; 4mhz xtal
BIT0_L      EQU    %11111110
LIGHT_BIT   EQU    %00000010
DLY01S      EQU    50
RAM     EQU    50H
;*********************************
;I/O PORT
PA      EQU    0
CS93C46     EQU    0            ;OUT
LIGHT_RELAY EQU    1            ;OUT
UP_RELAY    EQU    2
DW_RELAY    EQU    3
HIGH_LIMIT  EQU    4            ;IN
LOW_LIMIT   EQU    5            ;IN
MRT_SEL     EQU    6            ;IN
RF_IN       EQU    7            ;IN
;-------------------------------------
PB      EQU    1
SDO     EQU    5
SDI     EQU    6
SCLK        EQU    7
;-------------------------------------
PC      EQU    2
UP_DOOR_DEC EQU    0
DW_DOOR_INC EQU    1
DW_DOOR_DEC EQU    2
STORE_EEP   EQU    4
ADO     EQU    6
;-------------------------------------
PD      EQU    3
UP_DOOR_INC EQU    5
;-------------------------------------
PA_DIR      EQU    4            ;0=IN 1=OUT
PB_DIR      EQU    5            ;0=IN 1=OUT;ONLY BIT 5,6,7
PC_DIR      EQU    6            ;0=IN 1=OUT BIT 0,1,2 I/O 3,4,5,6 A/D
PD_DIR      EQU    7            ;0=IN 1=OUT BIT 5
;*********************************
; RAM 50H-0BFH
;*********************************
        ORG    RAM
```

```
TMP0        DS    1
TMP1        DS    1
TMP2        DS    1
;*****************************
FLAG        DS    1
TIM_2MS     EQU   0
FLASH_FG    EQU   1
STORE_FG    EQU   2
LIGHT_ON    EQU   3
RF_RECOK    EQU   4
ASSIGN_DOWN EQU   6
EEP_CLRFG   EQU   7
;*****************************
MOTOR_FG    DS    1
DIR_FG      EQU   0          ;1=UP 0=DOWN
IR_ERROR    EQU   1
LOW_SPEED   EQU   2
MRT_OVER    EQU   3
SAMP_OK     EQU   4
MOTOR_ON    EQU   5
HZ_50       EQU   6
;*****************************
ERROR_FG    DS    1
MRT_ERR     EQU   0
RPM_ERR     EQU   1
IR_ERR      EQU   2
ASS_DOWN    EQU   3
SEQU_FG     EQU   4
SAMP_AD     EQU   5
;*****************************
SEC_CNT     DS    1
LIGHT_CNT0  DS    1
LIGHT_CNT1  DS    1

EEP_ADDR    DS    1
EEP_DATA0   DS    1
EEP_DATA1   DS    1
EEP_CNT     DS    1
EEP_BUF     DS    1
EEP_BUF1    DS    1
TIM_TMP0    DS    1
TIM_TMP1    DS    1
CNT0        DS    1
```

```
KEY_BUFF    DS    1
KEY_CNT     DS    1

UP_RANG     DS    1
DOWN_RANG   DS    1

MOTOR_CNT0  DS    1          ;CONTER 0.1SEC
MRT_CNT0    DS    1          ;20.28SEC
MRT_CNT1    DS    1          ;
STP_MTR_CNT DS    1          ;1.7SEC
PASS_MTR    DS    1
PASS_IR     DS    1
WAIT_START  DS    1

IR_CNT      DS    1

RPM_TMP0    DS    1
RPM_TMP1    DS    1

RPM_CAL0    DS    1
RPM_CAL1    DS    1

RPM_CNT0    DS    1
RPM_CNT1    DS    1
RPM_CNT2    DS    1
SENSOR_CNT  DS    1

REC_DATA0   DS    1
REC_DATA1   DS    1
REC_DATA2   DS    1
REC_DATA3   DS    1
REC_CNT     DS    1

ASSIGN_CNT  DS    1

RF_OLD_CNT  DS    1
RF_OLD_CNT1 DS    1
;*********************************
LIMIT_SW    DS    1
LIMIT_CNT   DS    1

RTMP0       DS    1
```

```
RTMP1       DS      1
ADC_TMP     DS      1
ADC_TMP1    DS      1
;*********************************
        ORG     100H



RESET:
        SEI
        RSP             ;CLEAR STACK POINT
        JSR     DELAY_1S
        JSR     CLEAR_RAM
        JSR     INIT_IO
        JSR     INIT_TIME
        JSR     INIT_LIGHT
        JSR     READ_RANG
        CLI
;*********************************************
MAIN:
        LDA     #0
        STA     COPC
        JSR     TEST_RPMS
        BRCLR       SAMP_AD,ERROR_FG,NO_RPMS
        BCLR        SAMP_AD,ERROR_FG
        JSR     READBOX
NO_RPMS:
        LDA     #2
        STA     TMP1
        CLR     TMP0
SER_NOISE:
        BRSET       RF_IN,PA,NOISE      ;
        DEC     TMP0            ;
        BNE     SER_NOISE           ; NOISE 3MS
        DEC     TMP1            ;
        BNE     SER_NOISE           ; NOISE 3MS
;*********************************
        LDA     #2
        STA     TMP1
SER_TOTAL:
        BRSET       RF_IN,PA,START_DATA      ;
        DEC     TMP0
        BNE     SER_TOTAL
```

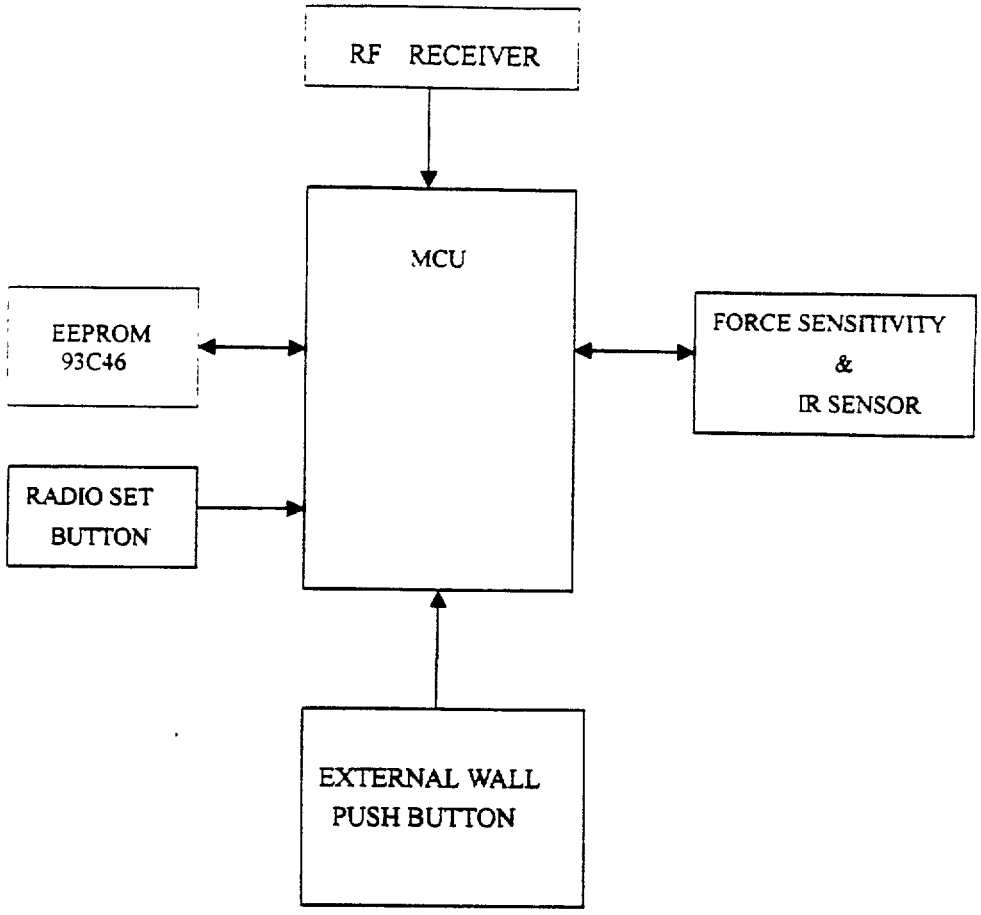
```
        DEC     TMP1
        BNE     SER_TOTAL
        BEQ     NOISE
;*****************************
START_DATA:
        JSR     RF_REC
        BRCLR       RF_RECOK,FLAG,NOISE
        BRSET       STORE_FG,FLAG,STORE_PASS    ;IF MOTOR ON
                                ;NO SAVE PASSWORD
        JSR     CHK_PASSWORD
        BRCLR       RF_RECOK,FLAG,NOISE
        BCLR        RF_RECOK,FLAG
ANY_PNT:
        BRSET       SEQU_FG,ERROR_FG,RF_SEQU    ;SEUQR ON
        BRCLR       IR_ERR,ERROR_FG,REM_FUNC
        BRCLR       HIGH_LIMIT,LIMIT_SW,DSP_LIGHT
        BCLR        DIR_FG,MOTOR_FG
        JSR     MOTOR_PROC              ;5/27
DSP_LIGHT:
        JSR     ERROR_LIGHT
        JMP     MAIN
REM_FUNC:
        JSR     MOTOR_PROC
NOISE:
        JMP     MAIN
RF_SEQU:
        LDA     #10
        JSR     STORE_LIGHT
        JMP     MAIN
;*****************************
STORE_PASS:
        SEI
        BCLR        RF_RECOK,FLAG
;        BCLR        STORE_FG,FLAG
        LDX     #10
STORE_LOP:
        TXA
        JSR     READ_EEP1   ;
        INC     EEP_ADDR
        INC     EEP_ADDR
        JSR     WRITE_EEP   ;
        DECX                ;
        CPX     #0FFH
```

CONFIDENTIAL
LX 1117

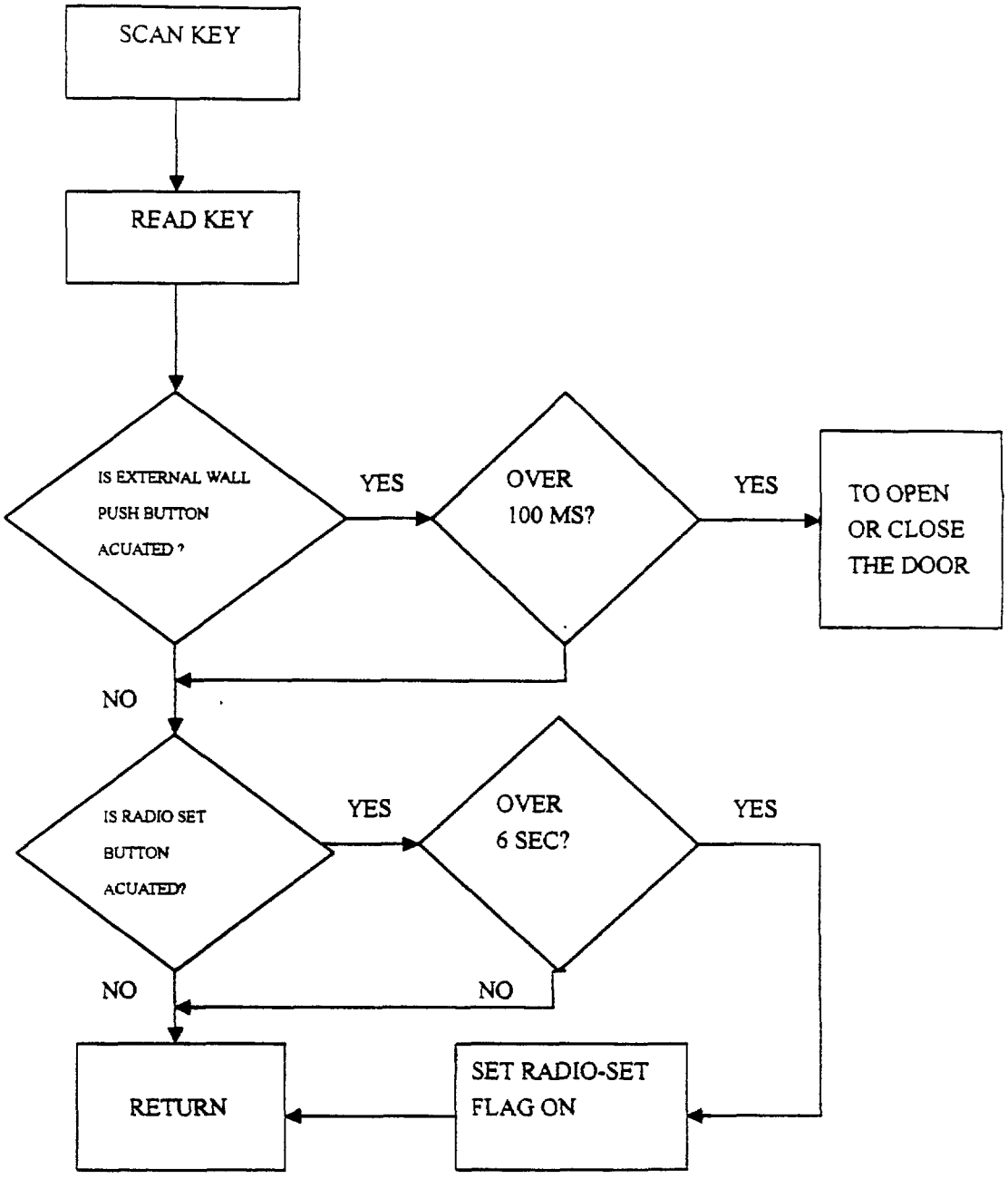
```
        BNE     STORE_LOP   ;
;-----------------------------------
        CLR     EEP_ADDR
        LDA     REC_DATA0
        STA     EEP_DATA0
        LDA     REC_DATA1
        STA     EEP_DATA1
        JSR     WRITE_EEP   ;
        INC     EEP_ADDR
        LDA     REC_DATA2
        STA     EEP_DATA0
        LDA     REC_DATA3
        STA     EEP_DATA1
        JSR     WRITE_EEP   ;
;*************************************
        CLR     EEP_ADDR
        JSR     READ_EEP    ;
        LDA     REC_DATA0
        CMP     EEP_DATA0
        BNE     VERY_ERROR
        LDA     REC_DATA1
        CMP     EEP_DATA1
        BNE     VERY_ERROR

        INC     EEP_ADDR
        JSR     READ_EEP    ;

        LDA     REC_DATA2
        CMP     EEP_DATA0
        BNE     VERY_ERROR
        LDA     REC_DATA3
        CMP     EEP_DATA1
        BNE     VERY_ERROR
;       JSR     ON_LIGHT
        LDA     #2
        JSR     STORE_LIGHT
        CLI
        JMP     MAIN
;*************************************
VERY_ERROR:
        JSR     ERROR_LIGHT
        CLI
        JMP     MAIN
```

CONFIDENTIAL
LX 1118

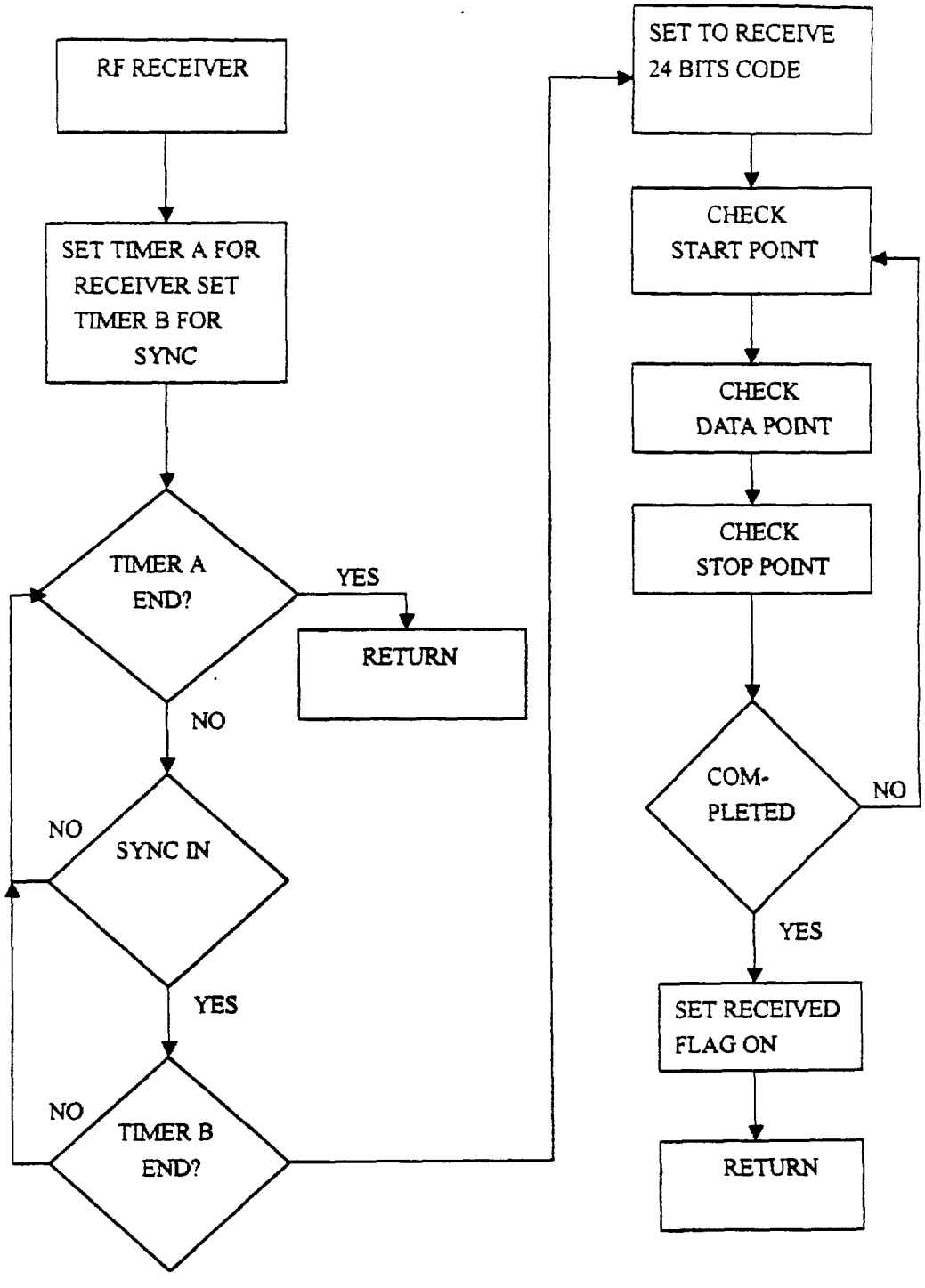
```
;*********************************
        INCLUDE   "UTIL.ASM"
        INCLUDE   "TIMER.ASM"
        INCLUDE   "EEP.ASM"
        INCLUDE   "LIGHT.ASM"
        INCLUDE   "KEYBOARD.ASM"
        INCLUDE   "AD_CON.ASM"
        INCLUDE   "MOTOR.ASM"
        INCLUDE   "IRQ.ASM"
        INCLUDE   "RF_REC.ASM"
;*********************************
;MOR     WDOG ENABLE
        ORG     1EFFH
        DB      0               ;
        DB      80H
;CLEAR WDOG WRITE 0 TO COPC
        ORG     1FF0H
COPC        DS      1

        ORG     $1FF8
        DW      TIMER_IRQ
        DW      IRQ_VECTOR
        DW      SWI_VECTOR
        DW      RESET

        END
```


RF RECEIVER
MCU
EEPROM
93C46
FORCE SENSITIVITY
&
IR SENSOR
RADIO SET
BUTTON
EXTERNAL WALL
PUSH BUTTON

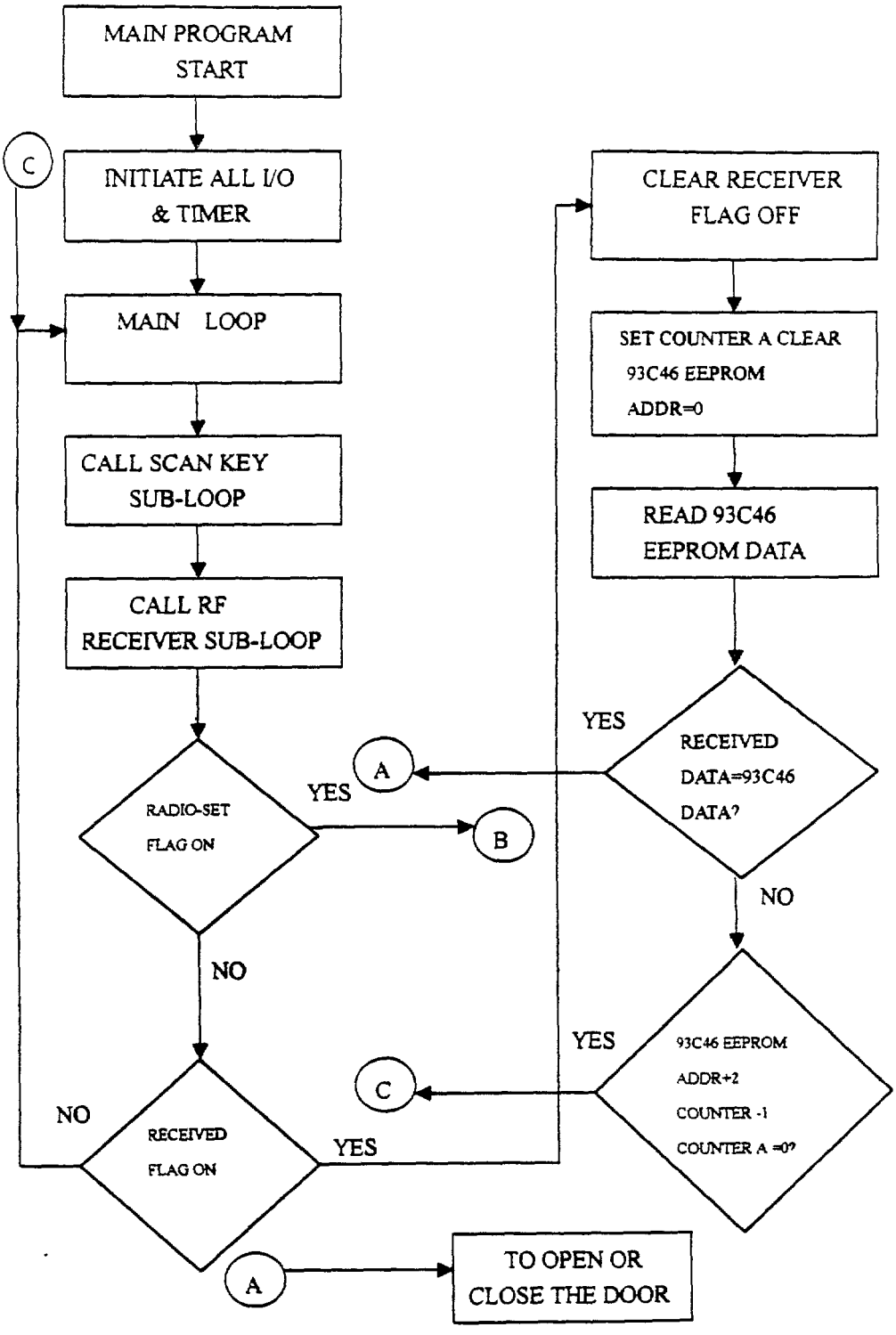
EXTERNAL WALL PUSH BUTTON & RADIO SET
BUTTON SCAN KEY CALL INTERRUPT SUB - LOOP



RF RECEIVER CALL INTERRUPT SUB-LOOP

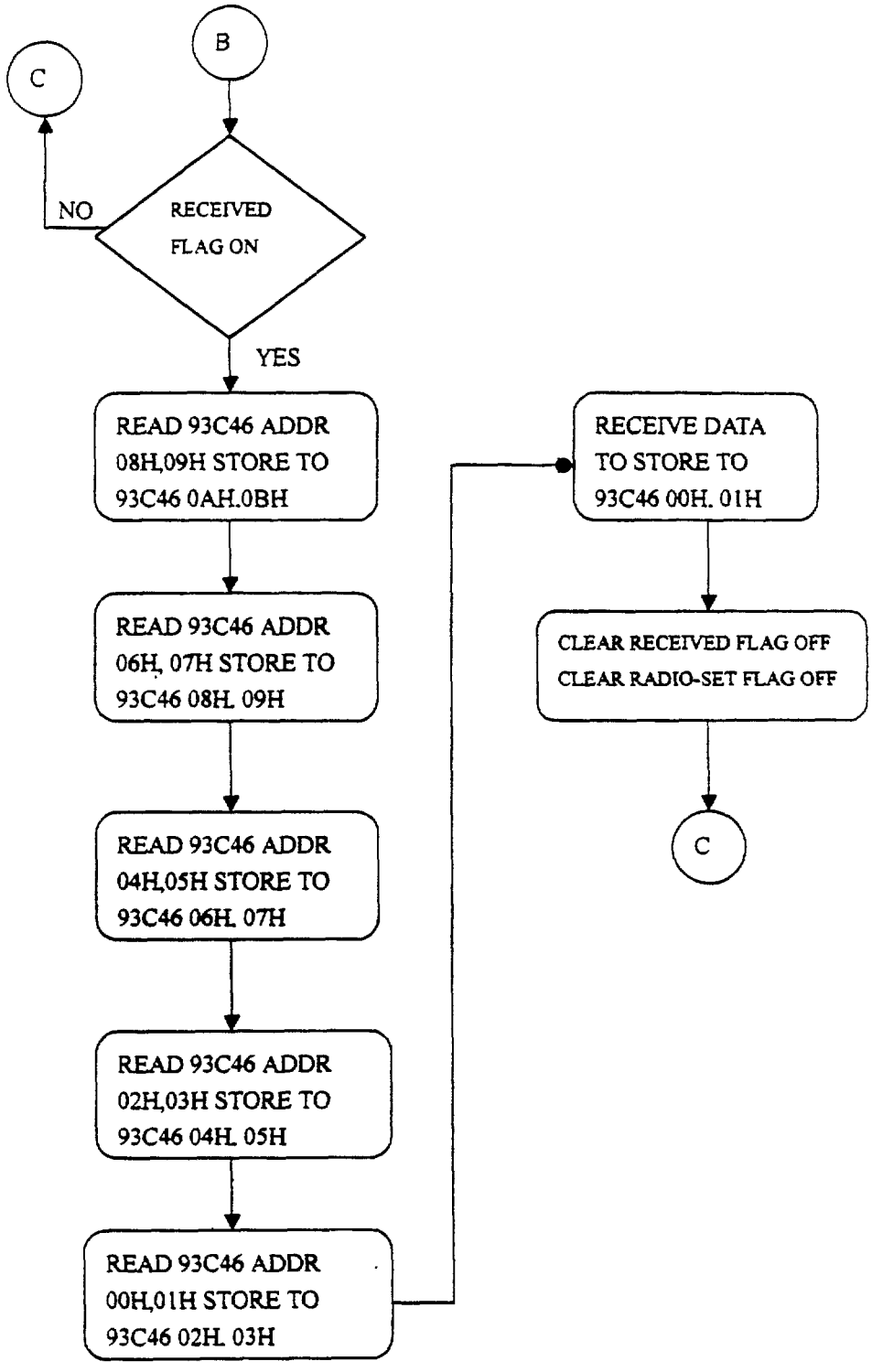


MAIN LOOP

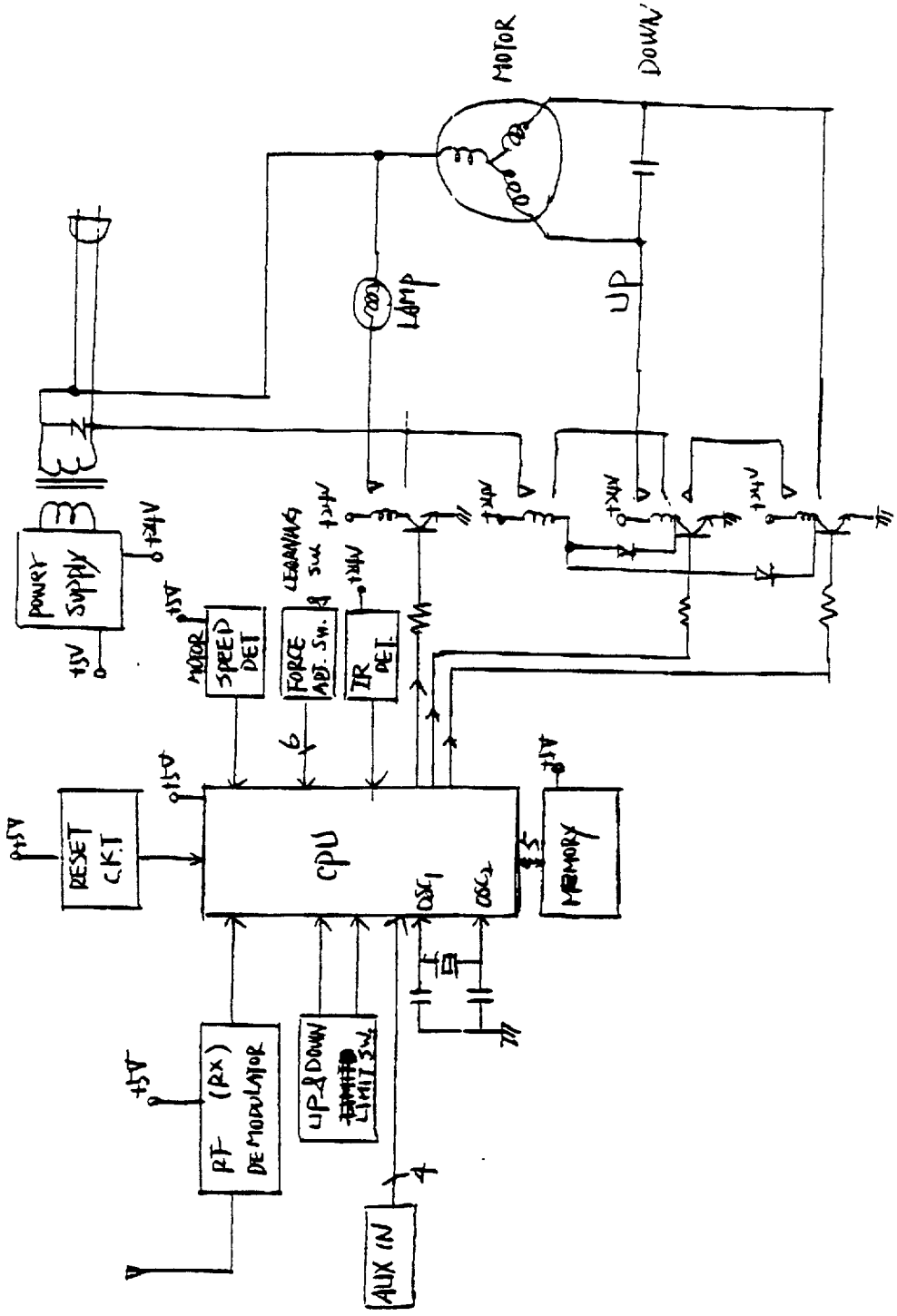


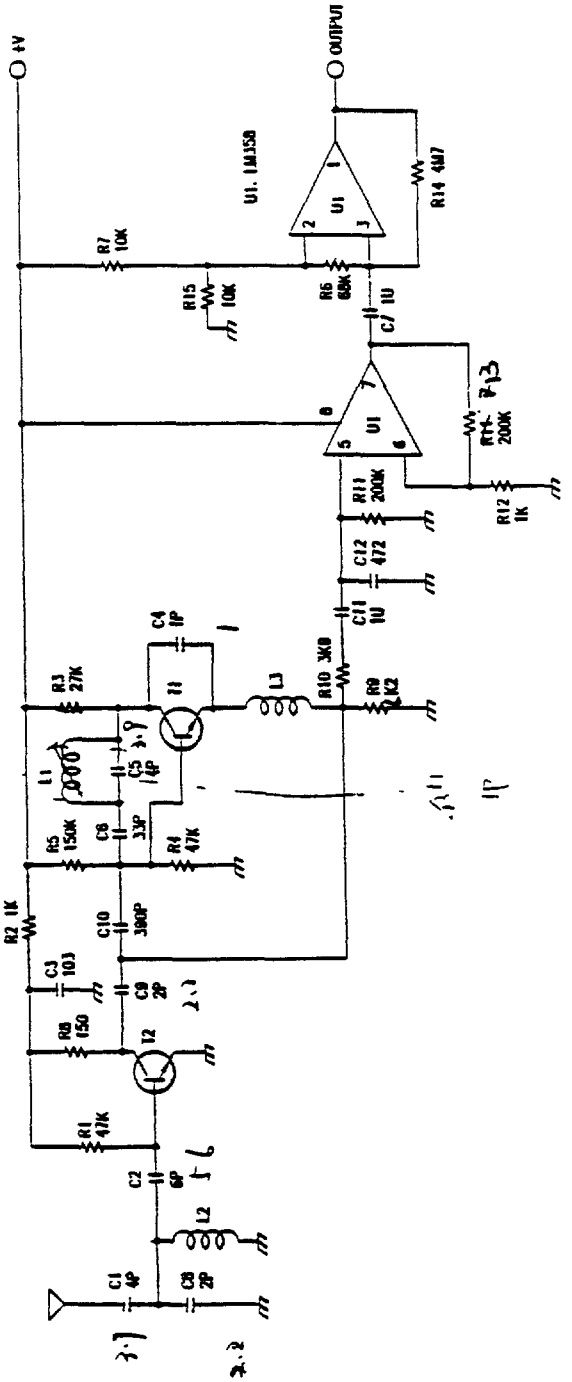
CONFIDENTIAL
LX 1123

RADIO SET SUB-LOOP
(CODE STORAGE)

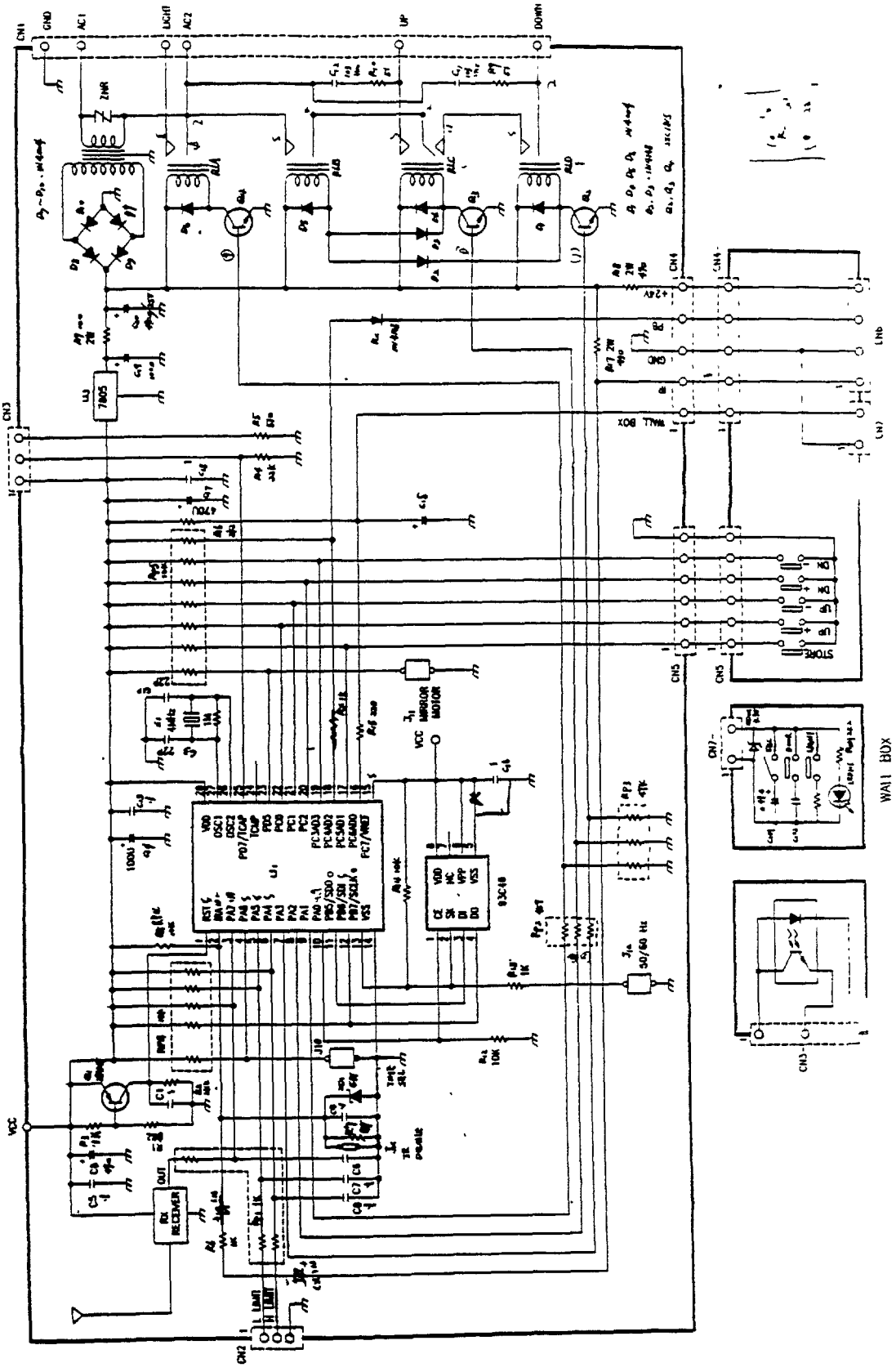


CONFIDENTIAL
LX 1124


MOTOR
DOWN
UP
LAMP
POWER SUPPLY
+5V
+12V
MOTOR SPEED DET
LEARNING SW.
FORCE ADJ. SW.
IR DET.
RESET C.K.T
CPU
OSC1
OSC2
MEMORY
RF (RX) DEMODULATOR
UP-DOWN LIMIT SW.
AUX IN



CONFIDENTIAL
LX 1126



CONFIDENTIAL
LX 1128

MICROCHIP

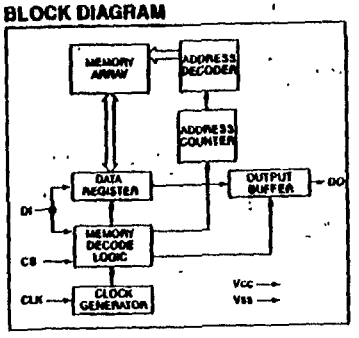
93C46B

1K 5.0V Microwire® Serial EEPROM

FEATURES

- Single supply 5.0V operation
- Low power CMOS technology
  - 1 mA active current (typical)
  - 1 µA standby current (maximum)
- 64 x 16 bit organization
- Self-timed ERASE and WRITE cycles (including auto-erase)
- Automatic ERAL before WRAL
- Power on/off data protection circuitry
- Industry standard 3-wire serial interface
- Device status signal during ERASE/WRITE cycles
- Sequential READ function
- 1,000,000 E/W cycles guaranteed
- Data retention > 200 years
- 8-pin PDIP/SOIC and 8-pin TSSOP packages
- Available for the following temperature ranges:
  - Commercial (C): 0°C to +70°C
  - Industrial (I): -40°C to +85°C
  - Automotive (E): -40°C to +125°C

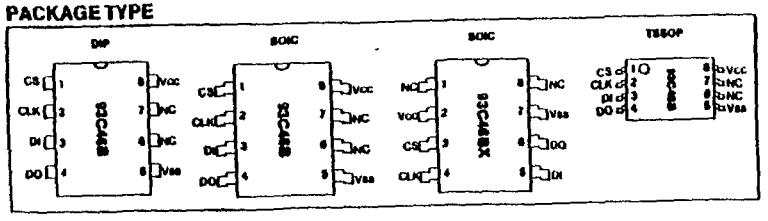
BLOCK DIAGRAM



DESCRIPTION

The Microchip Technology Inc 93C46B is a 1K-bit, low-voltage serial Electrically Erasable PROM. The device memory is configured as 64 x 16 bits. Advanced CMOS technology makes this device ideal for low power, nonvolatile memory applications. The 93C46B is available in standard 8-pin DIP, surface mount SOIC, and TSSOP packages. The 93C46BX are only offered in a 150 mil SOIC package.

PACKAGE TYPE



Microwire is a registered trademark of National Semiconductor Incorporated

LX 1129

1.0 ELECTRICAL CHARACTERISTICS

1.1 Maximum Ratings*

Vcc	7.0V
All inputs and outputs w.r.t. Vss	-0.6V to Vcc +1.0V
Storage temperature	-65°C to +150°C
Ambient temp. with power applied	-65°C to +125°C
Soldering temperature of leads (10 seconds)	+300°C
ESD protection on all pins	4 kV

*Notice: Stresses above those listed under "Maximum ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at those or any other conditions above those indicated in the operational listings of this specification is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

TABLE 1-1 PIN FUNCTION TABLE

Name	Function
CS	Chip Select
CLK	Serial Data Clock
DI	Serial Data Input
DO	Serial Data Output
Vss	Ground
NC	No Connect
Vcc	Power Supply

TABLE 1-2 DC AND AC ELECTRICAL CHARACTERISTICS

All parameters apply over the specified operating ranges unless otherwise noted	Commercial (C) Vcc = +4.5V to +5.5V Tamb = 0°C to +70°C Industrial (I) Vcc = +4.5V to +5.5V Tamb = -40°C to +85°C Automotive (E) Vcc = +4.5V to +5.5V Tamb = -40°C to +125°C				
Parameter	Symbol	Min.	Max	Units	Conditions
High level input voltage	VIH	2.0	Vcc +1	V	(Note 2)
Low level input voltage	VIL	-0.3	0.8	V	
Low level output voltage	VOL	—	0.4	V	IOL = 2.1 mA Vcc = 4.5V
High level output voltage	VOH	2.4	—	V	IOH = -400 µA Vcc = 4.5V
Input leakage current	ILI	-10	10	µA	VIN = Vss to Vcc
Output leakage current	ILO	-10	10	µA	VOUT = Vss to Vcc
Pin capacitance (all inputs/outputs)	CIN, COUT	—	7	pF	VIN/VOUT = 0 V (Notes 1 & 2) Tamb = +25°C FCLK = 1 MHz
Operating current	ICC read	—	1	mA	
	ICC write	—	1.5	mA	
Standby current	ICCS	—	1	µA	CS = Vss DI = Vss
Clock frequency	FCLK	—	2	MHz	Vcc = 4.5V
Clock high time	TCKH	250	—	ns	
Clock low time	TCKL	250	—	ns	
Chip select setup time	TCSS	50	—	ns	Relative to CLK
Chip select hold time	TCSH	0	—	ns	Relative to CLK
Chip select low time	TCSL	250	—	ns	
Data input setup time	TDIS	100	—	ns	Relative to CLK
Data input hold time	TDIH	100	—	ns	Relative to CLK
Data output delay time	TPD	—	400	ns	CL = 100 pF
Data output disable time	TCZ	—	100	ns	CL = 100 pF (Note 2)
Status valid time	TSV	—	500	ns	CL = 100 pF
Program cycle time	TWC	—	2	ms	ERASE/WRITE mode
	TEC	—	6	ms	ERAL mode
	TWL	—	15	ms	WRAL mode
Endurance	—	1M	—	cycles	25°C Vcc = 5.0V Block Mode (Note 3)

Note 1: This parameter is tested at Tamb = 25°C and FCLK = 1 MHz

2 This parameter is periodically sampled and not 100% tested

3 This application is not tested but guaranteed by characterization. For endurance estimates in a specific application, please consult the Total Endurance Model which may be obtained on our website.

3.4 ERASE

The ERASE instruction forces all data bits of the specified address to the logical "1" state. This cycle begins on the rising clock edge of the last address bit.

The DO pin indicates the READY/BUSY status of the device if CS is brought high after a minimum of 250 ns low (T_{CSL}). DO at logical "0" indicates that programming is still in progress. DO at logical "1" indicates that the register at the specified address has been erased and the device is ready for another instruction.

3.5 Erase All (ERAL)

The Erase All (ERAL) instruction will erase the entire memory array to the logical "1" state. The ERAL cycle is identical to the ERASE cycle, except for the different opcode. The ERAL cycle is completely self-timed and commences at the rising clock edge of the last address bit. Clocking of the CLK pin is not necessary after the device has entered the ERAL cycle.

The DO pin indicates the READY/BUSY status of the device, if CS is brought high after a minimum of 250 ns low (T_{CSL}) and before the entire ERAL cycle is complete.

FIGURE 3-2: ERASE TIMING

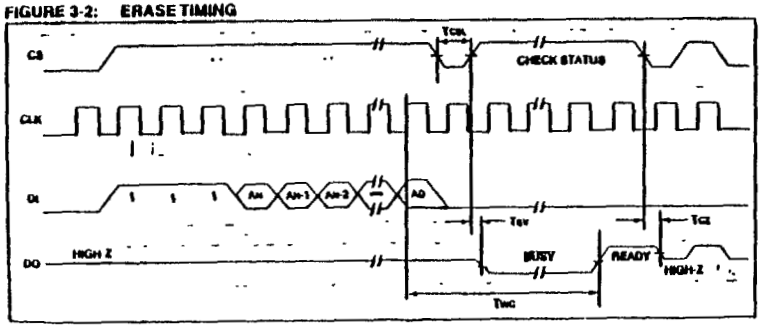
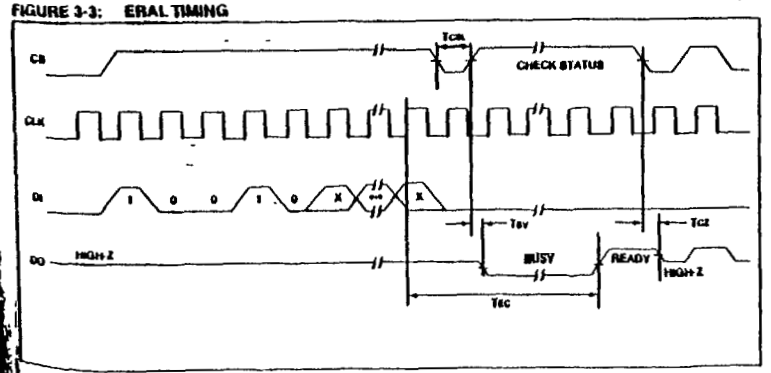


FIGURE 3-3: ERAL TIMING



3.6 ERASE/WRITE Disable and Enable (EWDS/EWEN)

The device powers up in the ERASE/WRITE Disable (EWDS) state. All programming modes must be preceded by an Erase/Write Enable (EWEN) instruction. Once the EWEN instruction is executed, programming remains enabled until an EWDS instruction is executed or Vcc is removed from the device. To protect against accidental data disturbance, the EWDS instruction can be used to disable all ERASE/WRITE functions and should follow all programming operations. Execution of a READ instruction is independent of both the EWDS and EWEN instructions.

3.7 READ

The READ instruction outputs the serial data of the addressed memory location on the DO pin. A dummy zero bit precedes the 16-bit output string. The output data bits will toggle on the rising edge of the CLK and are stable after the specified time delay (T_{PD}). Sequential read is possible when CS is held high. The memory data will automatically cycle to the next register and output sequentially.

FIGURE 3-4: EWDS TIMING

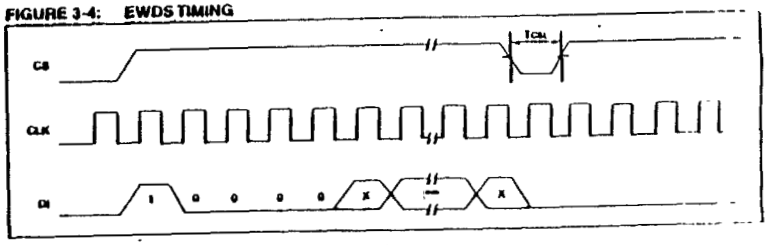


FIGURE 3-5: EWEN TIMING

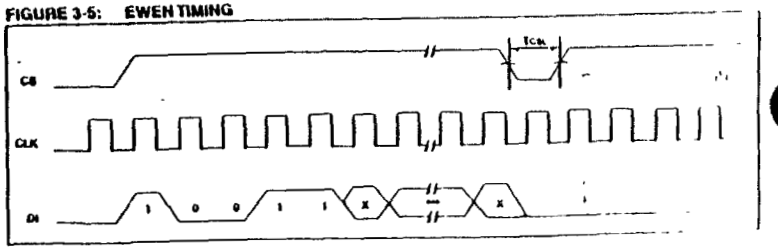
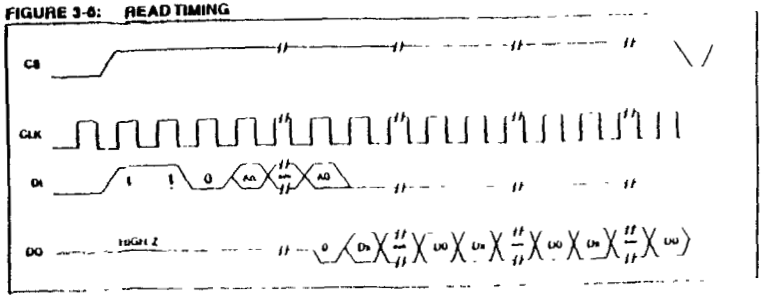


FIGURE 3-6: READ TIMING



2.0 PIN DESCRIPTION

2.1 Chip Select (CS)

A high level selects the device, a low level deselects the device and forces it into standby mode However, a programming cycle which is already in progress will be completed, regardless of the Chip Select (CS) input signal. If CS is brought low during a program cycle, the device will go into standby mode as soon as the programming cycle is completed.

CS must be low for 250 ns minimum (TCSL) between consecutive instructions If CS is low, the internal control logic is held in a RESET status.

2.2 Serial Clock (CLK)

The Serial Clock (CLK) is used to synchronize the communication between a master device and the 93C46B Opcodes, addresses, and data bits are clocked in on the positive edge of CLK Data bits are also clocked out on the positive edge of CLK.

CLK can be stopped anywhere in the transmission sequence (at high or low level) and can be continued anytime with respect to clock high time (TCKH) and clock low time (TCKL). This gives the controlling master freedom in preparing the opcode, address, and data

CLK is a "Don't Care" if CS is low (device deselected) If CS is high, but START condition has not been detected, any number of clock cycles can be received by the device, without changing its status (i e , waiting for a START condition)

CLK cycles are not required during the self-timed WRITE (i.e , auto ERASE/WRITE) cycle

After detecting a START condition, the specified number of clock cycles (respectively low to high transitions of CLK) must be provided These clock cycles are required to clock in all required opcodes, addresses, and data bits before an instruction is executed (Table 2-1) CLK and DI then become don't care inputs waiting for a new START condition to be detected

[illegible] CS must go low between consecutive instructions.

2.3 Data In (DI)

Data In (DI) is used to clock in a START bit, opcode, address, and data synchronously with the CLK input.

2.4 Data Out (DO)

Data Out (DO) is used in the READ mode to output data synchronously with the CLK input (TPD after the positive edge of CLK)

This pin also provides READY/BUSY status information during ERASE and WRITE cycles READY/BUSY status information is available on the DO pin if CS is brought high after being low for minimum chip select low time (TCSL) and an ERASE or WRITE operation has been initiated.

The status signal is not available on DO, if CS is held low during the entire ERASE or WRITE cycle In this case, DO is in the HIGH-Z mode If status is checked after the ERASE/WRITE cycle, the data line will be high to indicate the device is ready

TABLE 2-1 INSTRUCTION SET FOR 93C46B

Instruction	SB	Opcode	Address						Data In	Data Out	Req. CLK Cycles
ERASE	1	11	A5	A4	A3	A2	A1	A0	—	(RDY/BSY)	9
ERAL	1	00	1	0	X	X	X	X	—	(RDY/BSY)	9
EWDS	1	00	0	0	X	X	X	X	—	HIGH Z	9
EWEN	1	00	1	1	X	X	X	X	—	HIGH Z	9
READ	1	10	A5	A4	A3	A2	A1	A0	—	D15 - D0	25
WRITE	1	01	A5	A4	A3	A2	A1	A0	D15 - D0	(RDY/BSY)	25
WRAL	1	00	0	1	X	X	X	X	D15 - D0	(RDY/BSY)	25

LX 1130

4

Microwire®

93C46B

3.0 FUNCTIONAL DESCRIPTION

Instructions addresses and write data are clocked into the DI pin on the rising edge of the clock (CLK) The DO pin is normally held in a HIGH Z state except when reading data from the device, or when checking the READY/BUSY status during a programming operation The READY/BUSY status can be verified during an ERASE/WRITE operation by polling the DO pin; DO low indicates that programming is still in progress, while DO high indicates the device is ready The DO will enter the HIGH-Z state on the falling edge of the CS

3.1 START Condition

The START bit is detected by the device if CS and DI are both high with respect to the positive edge of CLK for the first time

Before a START condition is detected, CS, CLK, and DI may change in any combination (except to that of a START condition), without resulting in any device operation (ERASE, ERAL, EWDS, EWEN, READ, WRITE, and WRAL) As soon as CS is high, the device is no longer in the standby mode

An instruction following a START condition will only be executed if the required amount of opcodes, addresses, and data bits for any particular instruction is clocked in

After execution of an instruction (i e , clock in or out of the last required address or data bit) CLK and DI become don't care bits until a new START condition is detected

3.2 Data In (DI) and Data Out (DO)

It is possible to connect the Data In (DI)and Data Out (DO) pins together However, with this configuration if A0 is a logic high level, it is possible for a "bus conflict" to occur during the "dummy zero" that precedes the READ operation Under such a condition the voltage level seen at DO is undefined and will depend upon the relative impedances of DO and the signal source driving A0 The higher the current sourcing capability of A0 the higher the voltage at the DO pin

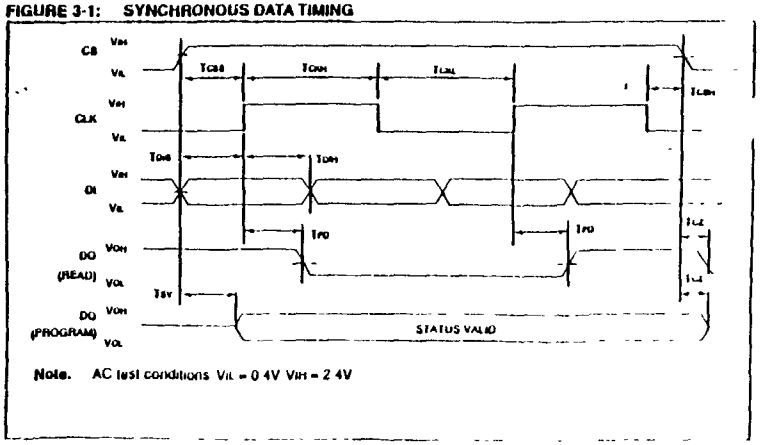
3.3 Data Protection

During power-up, all programming modes of operation are inhibited until Vcc has reached a level greater than 3 8V During power-down the source data protection circuitry acts to inhibit all programming modes when Vcc has fallen below 3 8V at nominal conditions

The ERASE/SRITE Disable (EWDS) and ERASE/WRITE Enable (EWEN) commands give additional protection against accidental programming during normal operation

After power-up, the device is automatically in the EWDS mode Therefore, an EWEN instruction must be performed before any ERASE or WRITE instruction can be executed

FIGURE 3-1: SYNCHRONOUS DATA TIMING



Note. AC test conditions VIL = 0 4V VIH = 2 4V

3.8 WRITE

The WRITE instruction is followed by 16 bits of data, which are written into the specified address. After the last data bit is clocked into the DI pin, the self-timed auto-erase and programming cycle begins.

The DO pin indicates the READY/BUSY status of the device, if CS is brought high after a minimum of 250 ns low (T_{CSL}) and before the entire write cycle is complete. DO at logical "0" indicates that programming is still in progress. DO at logical "1" indicates that the register at the specified address has been written with the data specified and the device is ready for another instruction.

3.9 Write All (WRAL)

The Write All (WRAL) instruction will write the entire memory array with the data specified in the command. The WRAL cycle is completely self-timed and commences at the rising clock edge of the last data bit. Clocking of the CLK pin is not necessary after the device has entered the WRAL cycle. The WRAL command does include an automatic ERAL cycle for the device. Therefore, the WRAL instruction does not require an ERAL instruction, but the chip must be in the EWEN status.

The DO pin indicates the READY/BUSY status of the device if CS is brought high after a minimum of 250 ns low (T_{CSL}).

FIGURE 3-7: WRITE TIMING

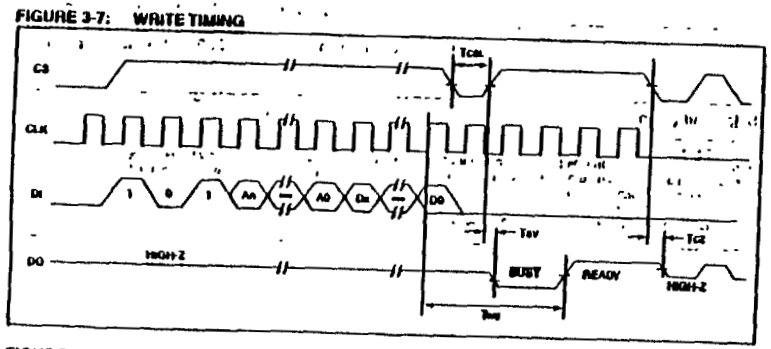
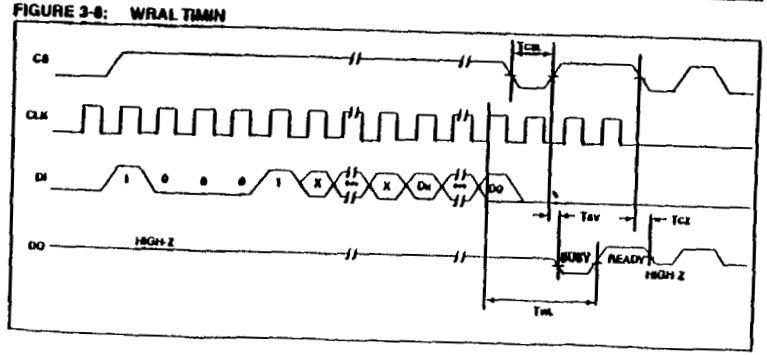


FIGURE 3-8: WRAL TIMIN



93C46B PRODUCT IDENTIFICATION SYSTEM

To order or obtain information, e.g., on pricing or delivery, refer to the factory or the listed sales office.

93C46B	—	/P	
		Package:	P = Plastic DIP (300 mil Body) 8 lead
			SN = Plastic SOIC (150 mil Body), 8-lead
			SM = Plastic SOIC (208 mil Body), 8-lead
			ST = TSSOP, 8-lead
		Temperature Range:	Blank = 0°C to +70°C
			I = -40°C to +85°C
			E = -40°C to +125°C
		Device:	93C46B = 1K Microwire Serial EEPROM
			93C46BT = 1K Microwire Serial EEPROM Tape and Reel
			93C46BX = 1K Microwire Serial EEPROM in alternate pinout (SN only)
			93C46BXT = 1K Microwire Serial EEPROM in alternate pinout, Tape and Reel (SN only)

Sales and Support

Data Sheets

Products supported by a preliminary Data Sheet may have an errata sheet describing minor operational differences and recommended workarounds. To determine if an errata sheet exists for a particular device, please contact one of the following.

1. Your local Microchip sales office
2. The Microchip Corporate Literature Center U.S. FAX (602) 786 7277
3. The Microchip Worldwide Web Site (www.microchip.com)

HOLTEK

HT6P20
2^{24} OTP Encoder

Features

- Operating voltage 2V~12V
- Low power consumption
- Built-in oscillator needs only 5% resistor
- 0/2/4/8 data selectable
- 2^{24} maximum address and data codes
- Easy interface with an RF or IR medium
- One time programmable process
- Data active: D0~D7
- Minimal external components

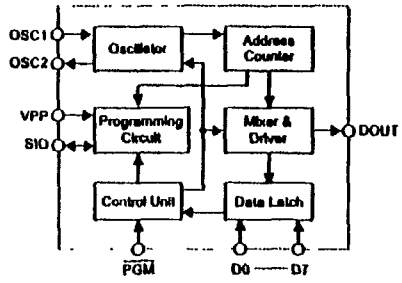
Applications

- Burglar alarm system
- Smoke and fire alarm system
- Garage door controllers
- Car door controllers
- Security system
- Cordless telephones
- Other remote control systems

General Description

The HT6P20 is a CMOS LSI encoder designed for remote control system applications. It encodes 24 bits of information and then serially transmits it via the DOUT pin upon receipt of transmission enable (DATA pins: D0~D7) signals. The combination of address and data bits of the HT6P20 is designed using one time programmable process. In addition, the chip offers various packaging for flexible combination of programmable address/data so as to meet various applications. Its programmable address/data is transmitted together with the anti-code bits via RF or infrared transmission medium upon receipt of a trigger signal.

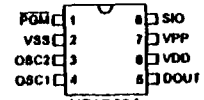
Block Diagram



Note: Address/Data numbers are available in various combinations, refer to the functional description

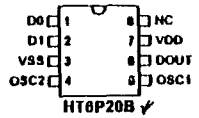
Pin Assignment

24-Address 0-Data



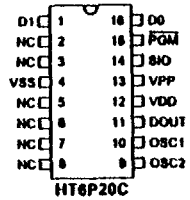
HT6P20A
– 8 DIP/NSOP

22-Address 2-Data



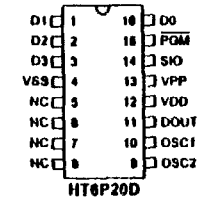
HT6P20B
– 8 DIP/NSOP

22-Address 2-Data



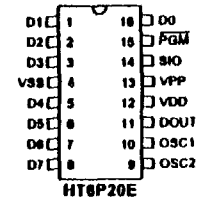
HT6P20C
– 16 DIP/NSOP

20-Address 4-Data



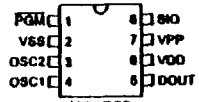
HT6P20D
– 16 DIP/NSOP

16-Address 8-Data



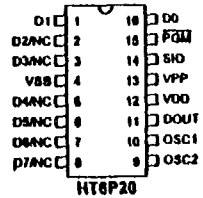
HT6P20E
– 16 DIP/NSOP

Blank device



HT6P20
– 8 DIP/NSOP

Blank device



HT6P20
– 16 DIP/NSOP

Notes: The customer code and control code have been programmed into HT6P20A/B/C/D/E Holtek's factory process.

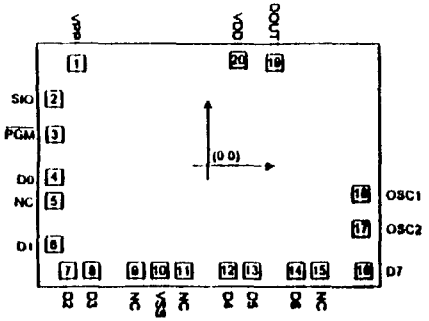
The blank device HT6P20 is not programmed, it can be programmed by Holtek's programming kit.

Warning: The 8 pin blank device only allows programming as HT6P20A type.

The 16-pin blank device allows programming as one of HT6P20C/D/E type. After programming the HT6P20 pin name are the same as one of HT6P20C/D/E, which is determined the kit programming procedure.

Pin name "NC" stands for no connection (floating).

Pad Assignment



Chip size: 2690 × 2010 (μm)^2

* The IC substrate should be connected to VSS in the PCB layout artwork

Pad Coordinates

Unit: μm

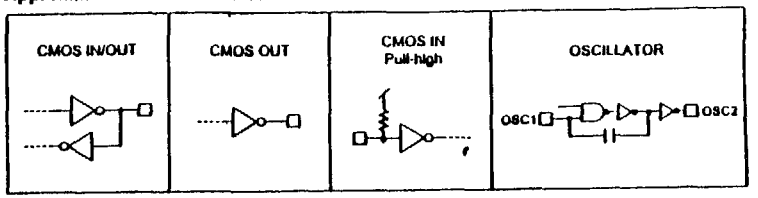
Pad No.	X	Y
1	-956 50	762 75
2	-1123 00	498 75
3	-1123 00	236 25
4	-1123 00	-86 25
5	-1123 00	-261 25
6	-1123 00	-583 75
7	-1027 50	-776 75
8	-852 50	-776 75
9	-530 00	-776 75
10	-355.00	-776 75
11	-180 00	-776 75
12	142 50	-776 75
13	317.50	-776 75
14	640 00	-776 75
15	815.00	-776 75
16	1137.50	-776 75
17	1118 00	-466 75
18	1118 00	-205 75
19	489 00	772 05
20	223 50	789 75

Pin Description

HT6P20E

Pin No	Pin Name	I/O	Internal Connection	Description
16 1~3	D0~D3	I	CMOS IN Pull-high	Data input and transmission enable (active low) They can be externally set to VSS or left open
4	VSS	I	—	Negative power supply (GND)
5~8	D4~D7	I	CMOS IN Pull-high	Data input and transmission enable (active low) They can be externally set to VSS or left open.
9	OSC2	O	OSCILLATOR	Oscillator output pin
10	OSC1	I	OSCILLATOR	Oscillator input pin
11	DOUT	O	CMOS OUT	Data serial transmission output
12	VDD	I	—	Positive power supply
13	VPP	I	—	Programming power supply, V_{DD} for normal operation
14	SIO	I/O	CMOS IN/OUT	Programming address/control code input and mode code output for mode verification
15	$\overline{PGM}$	I	CMOS IN Pull-high	Program mode control pin, active low

Approximate internal connection circuits



Absolute Maximum Ratings

Supply Voltage ..... -0 3V to 12V

Input Voltage..... V_{SS}-0 3V to V_{DD}+0.3V

Storage Temperature . . ... -50°C to 125°C

Operating Temperature . . -20°C to 75°C

Note. These are stress ratings only. Stresses exceeding the range specified under "Absolute Maximum Ratings" may cause substantial damage to the device Functional operation of this device at other conditions beyond those listed in the specification is not implied and prolonged exposure to extreme conditions may affect device reliability

Electrical Characteristics

Ta=25°C

Symbol	Parameter		Test Conditions V_{DD}	Test Conditions Conditions	Min.	Typ.	Max	Unit
V_{DD}	Operating Voltage		—	—	2	—	12	V
I_{STB}	Standby Current		12V	Oscillator stops	—	1	2	μA
I_{DD}	Operating Current		12V	No load f_{OSC}=3kHz	—	200	400	μA
V_{IH}	"H" Input Voltage		—	—	0 8V_{DD}	—	V_{DD}	V
V_{IL}	"L" Input Voltage		—	—	0	—	0 2V_{DD}	V
R_{PH}	D0~D7 Pull-high Resistance		12V	—	—	150	300	kΩ
I_{DOUT}	Output Current	Source	5V	0 9V_{DD}	-2	-5	—	mA
			12V	0 9V_{DD}	-6 5	-15	—	mA
		Sink	5V	0 1V_{DD}	2	5	—	mA
			12V	0.1V_{DD}	8	15	—	mA
f_{OSC}	Oscillator Frequency		12V	R_{OSC}=1 4MΩ	—	3	—	kHz

Functional Description

Programming

The HT6P20 is a one time programmable encoding integrated circuit The programmable codes consist of 4-bit control codes and 24-bit address codes Before using the device, first program the internal 4 control bits and 24 address bits Then, to program the address and control codes, connect the $\overline{PGM}$ pin to VSS, allowing the SIO pin to receive control and address codes from the continuously transmitting programming kit. The control codes (B0~B3) and address timing are respectively shown on the Table and the Figure for Programming Timing In the programming process, the mode bit (B0) has to be programmed first The 0/2/4/8 data number is selected by the address along with the data format bits (B1,B2) The HT6P20, in addition, provides a polarity bit (B3) to select the polarity of the output data.

Various packages of the HT6P20 offer programmable combinations of address/data to meet various application needs

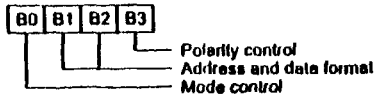


Table 1: Control code

• Mode control (B0)

B0	Function
0	Not programmed
1	Programmed

• Address and data format (B1,B2)

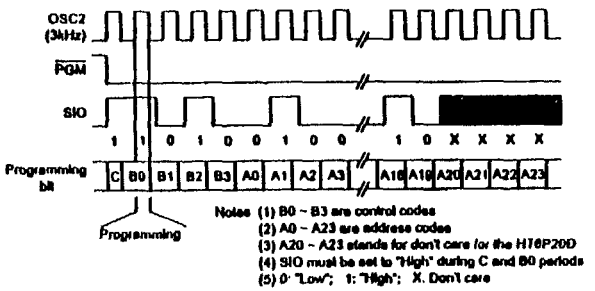
B2	B1	Code Format
0	0	A0~A23
0	1	A0~A21 + D1~D0
1	0	A0~A19 + D3~D0
1	1	A0~A15 + D7~D0

• Polarity control (B3)

B3	Data Output Polarity
0	Negative polarity
1	Positive polarity

The HT6P20A/C/D/E can be programmed by the SIO pins The HT6P20B, on the other hand, is programmed for 2-data and positive polarity application.

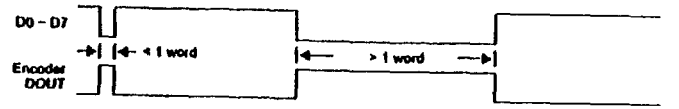
Programming timing of the HT6P20D (B0~B3=1010)


OSC2 (3kHz)
PGM
SIO
1 1 0 1 0 0 1 0 0 1 0 X X X X
Programming bit
C B0 B1 B2 B3 A0 A1 A2 A3 A16 A19 A20 A21 A22 A23
Programming
Notes (1) B0 ~ B3 are control codes
(2) A0 ~ A23 are address codes
(3) A20 ~ A23 stands for don't care for the HT6P20D
(4) SIO must be set to "High" during C and B0 periods
(5) 0: "Low"; 1: "High"; X: Don't care

Programming timing for the HT6P20D

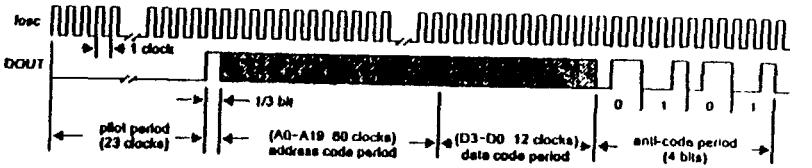
Normal operation

The HT6P20 encodes and transmits address/data to a decodor upon receipt of a trigger signal The address codes of the HT6P20A are always transmitted as long as power (VDD) is supplied The transmission function of the HT6P20B/C/D/E is enabled by the D0~D7 pins (active low) The following is the transmission timing of the HT6P20:



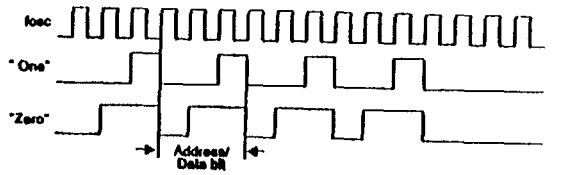
Transmission timing

A complete code word of the HT6P20D consists of 3 periods as shown below



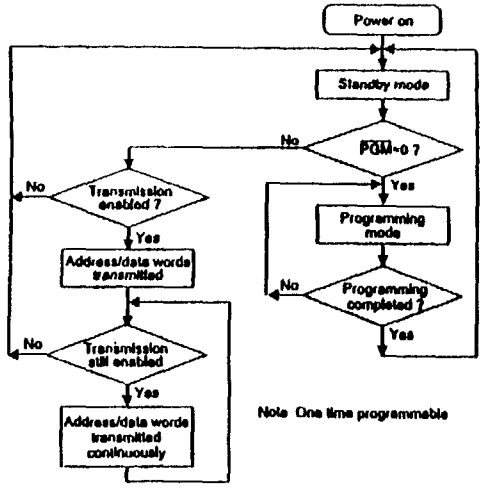
A complete code word for the HT6P20D

The HT6P20A/B/C/D/E detects the logic state of the internal programmed address and the external data pins, and then transmits the detected information during the code period Each address/data bit can be set to one of the following two logic states.



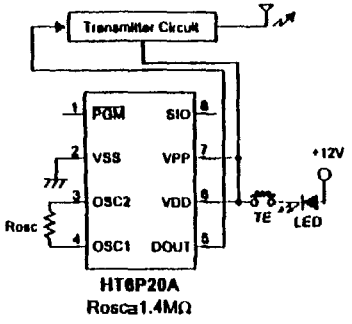
LX 1135

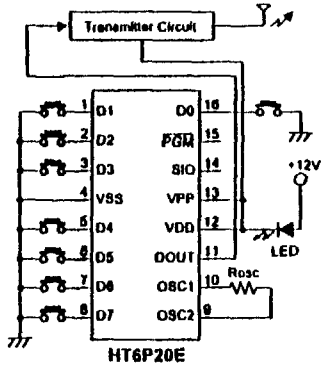
Flowchart

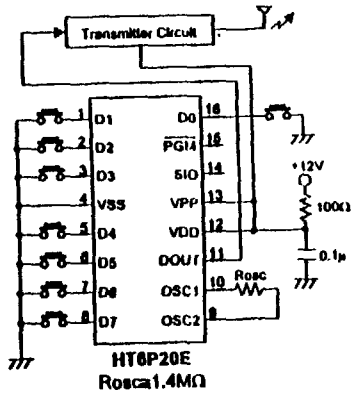


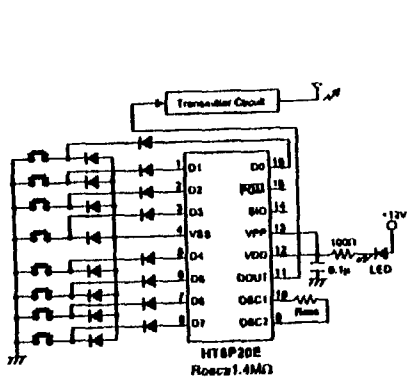
Note One time programmable

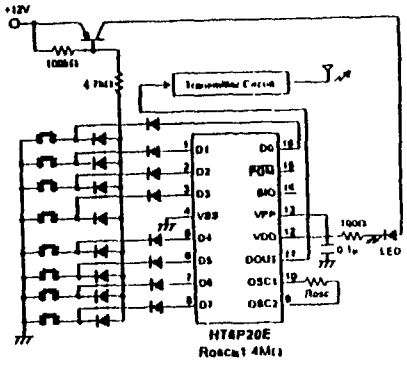
Application Circuits











Note: In order to prevent the IC from getting damaged due to the latch up, the 100Ω resistor or the LED which can also be a transmission indicator is indispensible when V_{DD}=9V~12V

LX 1136

```
ADIN            EQU         6
ADC_STATUS      EQU         1EH
ADC_ON          EQU         5
ADC_RC          EQU         6
CONV_FG         EQU         7
DEF_ADC_VAL     EQU         %01100000
ADC_DATA        EQU         1DH
;*********************************
READBOX:
                BCLR        ADIN,PC_DIR
                LDA         #10
                BRCLR       ADIN,PC,LOAD_LIGHT    ;TEST LIGHT

                BCLR        ADIN,PC                 ;DISCHING
                BSET        ADIN,PC_DIR
                LDA         #5
                JSR         DLY_100US   ;DISCH CAP   50US
                BCLR        ADIN,PC_DIR ;READ    SET IN READ
                LDA         #050
                BRCLR       ADIN,PC,LOAD_LIGHT    ;TEST PUSH BOX

                BCLR        ADIN,PC
                BSET        ADIN,PC_DIR    ;DISCH CAP

                LDA         #50
                JSR         DLY_100US   ;
                BCLR        ADIN,PC_DIR ;READ
                LDA         #090                  ;LOCK
                BRCLR       ADIN,PC,LOAD_LIGHT    ;TEST LOCK
                LDA         #120
LOAD_LIGHT:
                BSET        ADIN,PC
                BSET        ADIN,PC_DIR           ;CHG CAP 25US
                STA         ADC_TMP
;               CLI
                RTS
;*****************************
ANALY_ADC:
                STA         ADC_TMP1         ;SAVE_KEY
                LDA         #24          ;LIGHT
                CMP         ADC_TMP
                BCC         PUSH_LIGHT
                LDA         #70          ;PUSH BUTTEM
                CMP         ADC_TMP
                BCC         PUSH_PB
                LDA         #102         ;LOCK SEQU
                CMP         ADC_TMP
```


CONFIDENTIAL
LX 1138

```
                BCS         OPEN_SEQU
                BSET        SEQU_KEY,ADC_TMP1
                JMP         OPEN_SEQU
PUSH_PB
                BSET        PB_KEY,ADC_TMP1
OPEN_SEQU
PUSH_LIGHT
                LDA         ADC_TMP1
                RTS
```


CONFIDENTIAL
LX 1139

```
,   door program
; 4mhz xtal
$BASE 10T

DATA0           EQU         80H
DATA1           EQU         2DH
DATA_LONG       EQU         5
DATA_POS        EQU         16
RAM             EQU         50H
;*********************************
;I/O PORT
PA              EQU         0
IN_IRQ          EQU         0
START_SEND      EQU         1
;---------------------------------------
PB              EQU         1
SDO             EQU         5
SDI             EQU         6
SCLK            EQU         7
;---------------------------------------
PC              EQU         2
;---------------------------------------
PD              EQU         3
;---------------------------------------
PA_DIR          EQU         4           ;0=IN 1=OUT
PB_DIR          EQU         5           ;0=IN 1=OUT;ONLY BIT 5,6,7
PC_DIR          EQU         6           ;0=IN 1=OUT BIT 0,1,2 I/O 3,4,5,6
A/D
PD_DIR          EQU         7           ;0=IN 1=OUT BIT 5
;*********************************
SIO_CTRL        EQU         0AH
MASTER          EQU         4
SIOP_EN         EQU         6
;--------------------
SIO_STATUS      EQU         0BH
DATA_COLL       EQU         6
SPIF            EQU         7
SIO_DATA        EQU         0CH
, RAM 50H-0BFH
;*********************************
                ORG         RAM
TMP0            DS          1
TMP1            DS          1
TMP2            DS          1
;*****************************
```


CONFIDENTIAL
LX 1140

```
FLAG            DS          1
ERR_DATA        EQU         0
;*******************************
KEY_BUFF        DS          1
KEY_CNT         DS          1
REC_CNT         DS          1
CMP_CNT         DS          1
REC_INDEX       DS          1
CMP_INDEX       DS          1
REC_DATA0       DS          5
;*******************************
                ORG         60H
COMM_BANK       DS          40



;*******************************
                ORG         100H
POWER_ON:
                SEI
                RSP                         ;CLEAR STACK POINT
                LDA         #0FFH
                STA         MOR1
                LDA         #%00011000
                STA         MOR2
                JSR         DELAY_1S
                JSR         CLEAR_RAM
                JSR         INIT_IO
                LDA         #0
                STA         REC_CNT
MAIN:
                BRCLR       IN_IRQ,PA,$
                LDX         #0
                BRSET       IN_IRQ,PA,$
REC_LOP:
                LDA         #0
                STA         SIO_CTRL
                LDA         #%01000000
                STA         SIO_DATA
                STA         SIO_CTRL
                BRCLR       SPIF,SIO_STATUS,$
                LDA         SIO_DATA
                STA         REC_DATA0,X
                INCX
;               CPX         #DATA_LONG
                CPX         #5
                BNE         REC_LOP
```


CONFIDENTIAL
LX 1141

```
                BCLR    3,PA
                LDA     #80H
                CMP     REC_DATA0
                BNE     MAIN
;                LDA     #DATA0
;                CMP     REC_DATA0+3
;                BNE     MAIN
;                LDA     #DATA1
;                CMP     REC_DATA0+4
;                BNE     MAIN
;                BSET    3,PA_DIR
;                BSET    3,PA
;                JMP     MAIN

                LDA     #60H
                STA     CMP_INDEX
                LDA     #0
                STA     REC_INDEX
                STA     CMP_CNT
                STA     COPC
CMP_LOP:
                LDX     REC_INDEX
                LDA     REC_DATA0,X
                LDX     CMP_INDEX
                CMP     0,X
                BEQ     OLD_DATA
                BSET    ERR_DATA,FLAG
                JMP     NEX_DATA
OLD_DATA.
                INC     CMP_INDEX
                INC     REC_INDEX
                LDA     #DATA_LONG
                CMP     REC_INDEX
                BNE     CMP_LOP
                BCLR    ERR_DATA,FLAG
                JMP     MAIN
;------ COMP 1 LOP
NEX_DATA:
                INC     CMP_CNT
                LDA     CMP_CNT
                CMP     REC_CNT
                BCC     ANY_ERR
                LDA     #0
                STA     REC_INDEX
                LDA     CMP_CNT
                CLC
                ROLA
```


CONFIDENTIAL
LX 1142

```
                ROLA
                ROLA
,               INCA
                ADD         #60H
                STA         CMP_INDEX
                LDA         #DATA_POS
                CMP         CMP_CNT
                BNE         CMP_LOP
ANY_ERR.
                BRSET       ERR_DATA,FLAG,NEW_DATA
                JMP         MAIN
;               BRCLR       ERR_DATA,FLAG,MAIN
;--------------------------------

NEW_DATA:
                LDA         REC_CNT
                CLC
                ROLA
                ROLA
                ROLA
;               INCA
                ADD         #60H
                STA         CMP_INDEX
                LDA         #0
                STA         REC_INDEX
MOVE_LOP:
                LDX         REC_INDEX
                LDA         REC_DATA0,X
                LDX         CMP_INDEX
                STA         0,X
                INC         REC_INDEX
                INC         CMP_INDEX
                LDA         #DATA_LONG
                CMP         REC_INDEX
                BNE         MOVE_LOP
                INC         REC_CNT
                LDA         #DATA_POS
                CMP         REC_CNT
                BEQ         COMP_END
                JMP         MAIN
COMP_END:
                NOP
                JMP         COMP_END
;*******************************
CLEAR_RAM:
                LDX         #50H
CLR_LOP:
                CLR         0,X
```


CONFIDENTIAL
LX 1143

```
                INCX
                CPX         #0C0H
                BNE         CLR_LOP
                RTS
;*********************************
DELAY_1S.
                CLR         TMP0
                CLR         TMP1

DELAY_LOP:
                NOP
                NOP
                NOP
                DEC         TMP0
                BNE         DELAY_LOP
                LDA         #0
                STA         COPC
                DEC         TMP1
                BNE         DELAY_LOP
                RTS
;*********************************
INIT_IO:
                LDA         #0
                STA         PA
                LDA         #%00000000
                STA         PA_DIR

                LDA         #0
                STA         PB
                LDA         #%00100000
                STA         PB_DIR
                LDA         #0
                STA         PC_DIR          ;ALL IN
                LDA         #%00000000
                STA         PD_DIR          ;ALL OUT
                RTS
TIMER_IRQ:
IRQ_VECTOR:
SWI_VECTOR:
                RTI
;*********************************
                ORG         1EFFH
MOR1:
                DS          1
MOR2:
                DS          1
                ORG         1FF0H
COPC            DS          1
```


CONFIDENTIAL
LX 1144

```
ORG         $1FF8
DW          TIMER_IRQ
DW          IRQ_VECTOR
DW          SWI_VECTOR
DW          POWER_ON

END
```


CONFIDENTIAL
LX 1145

```
$BASE 10T
;   door program
; 4mhz xtal
BIT0_L          EQU     %11111110
LIGHT_BIT       EQU     %00000010
DLY01S          EQU     50
RAM             EQU     50H
;*********************************
;I/O PORT
PA              EQU     0
CS93C46         EQU     0               ;OUT
LIGHT_RELAY     EQU     1               ;OUT
UP_RELAY        EQU     2
DW_RELAY        EQU     3
HIGH_LIMIT      EQU     4               ;IN
LOW_LIMIT       EQU     5               ;IN
MRT_SEL         EQU     6               ;IN
RF_IN           EQU     7               ;IN
;---------------------------------------
PB              EQU     1
SDO             EQU     5
SDI             EQU     6
SCLK            EQU     7
;---------------------------------------
PC              EQU     2
UP_DOOR_DEC     EQU     0
DW_DOOR_INC     EQU     1
DW_DOOR_DEC     EQU     2
STORE_EEP       EQU     4
AD0             EQU     6
;---------------------------------------
PD              EQU     3
UP_DOOR_INC     EQU     5
;---------------------------------------
PA_DIR          EQU     4               ;0=IN 1=OUT
PB_DIR          EQU     5               ;0=IN 1=OUT;ONLY BIT 5,6,7
PC_DIR          EQU     6               ;0=IN 1=OUT BIT 0,1,2 I/O 3,4,5,6
A/D
PD_DIR          EQU     7               ;0=IN 1=OUT BIT 5
;*********************************
; RAM 50H-0BFH
;*********************************
                ORG     RAM
TMP0            DS      1
TMP1            DS      1
TMP2            DS      1
;*****************************
```


CONFIDENTIAL
LX 1146

```
FLAG            DS          1
TIM_2MS         EQU         0
FLASH_FG        EQU         1
STORE_FG        EQU         2
LIGHT_ON        EQU         3
RF_RECOK        EQU         4
ASSIGN_DOWN     EQU         6
EEP_CLRFG       EQU         7
;*****************************
MOTOR_FG        DS          1
DIR_FG          EQU         0           ;1=UP 0=DOWN
IR_ERROR        EQU         1
LOW_SPEED       EQU         2
MRT_OVER        EQU         3
SAMP_OK         EQU         4
MOTOR_ON        EQU         5
HZ_50           EQU         6
;*****************************
ERROR_FG        DS          1
MRT_ERR         EQU         0
RPM_ERR         EQU         1
IR_ERR          EQU         2
ASS_DOWN        EQU         3
SEQU_FG         EQU         4
SAMP_AD         EQU         5
;*****************************
SEC_CNT         DS          1
LIGHT_CNT0      DS          1
LIGHT_CNT1      DS          1

EEP_ADDR        DS          1
EEP_DATA0       DS          1
EEP_DATA1       DS          1
EEP_CNT         DS          1
EEP_BUF         DS          1
EEP_BUF1        DS          1
TIM_TMP0        DS          1
TIM_TMP1        DS          1
CNT0            DS          1

KEY_BUFF        DS          1
KEY_CNT         DS          1

UP_RANG         DS          1
DOWN_RANG       DS          1

MOTOR_CNT0      DS          1           ;CONTER 0.1SEC
MRT_CNT0        DS          1           ;20,28SEC
```


CONFIDENTIAL
LX 1147

```
MRT_CNT1        DS          1
STP_MTR_CNT     DS          1               ;1.7SEC
PASS_MTR        DS          1
PASS_IR         DS          1
WAIT_START      DS          1

IR_CNT          DS          1

RPM_TMP0        DS          1
RPM_TMP1        DS          1

RPM_CAL0        DS          1
RPM_CAL1        DS          1

RPM_CNT0        DS          1
RPM_CNT1        DS          1
RPM_CNT2        DS          1
SENSOR_CNT      DS          1

REC_DATA0       DS          1
REC_DATA1       DS          1
REC_DATA2       DS          1
REC_DATA3       DS          1
REC_CNT         DS          1

ASSIGN_CNT      DS          1

RF_OLD_CNT      DS          1
RF_OLD_CNT1     DS          1
;*********************************
LIMIT_SW        DS          1
LIMIT_CNT       DS          1

RTMP0           DS          1
RTMP1           DS          1
ADC_TMP         DS          1
ADC_TMP1        DS          1
;*********************************
                ORG         100H


RESET.
                SEI
                RSP                         ;CLEAR STACK POINT
                JSR         DELAY_1S
                JSR         CLEAR_RAM
                JSR         INIT_IO
```


CONFIDENTIAL
LX 1148

```
                JSR         INIT_TIME
                JSR         INIT_LIGHT
                JSR         READ_RANG
                CLI
;***********************************************
MAIN:
                LDA         #0
                STA         COPC
                JSR         TEST_RPMS
                BRCLR       SAMP_AD,ERROR_FG,NO_RPMS
                BCLR        SAMP_AD,ERROR_FG
                JSR         READBOX
NO_RPMS:
                LDA         #2
                STA         TMP1
                CLR         TMP0
SER_NOISE:
                BRSET       RF_IN,PA,NOISE      ;
                DEC         TMP0                ;
                BNE         SER_NOISE           ; NOISE 3MS
                DEC         TMP1                ;
                BNE         SER_NOISE           ; NOISE 3MS
;*******************************
                LDA         #2
                STA         TMP1
SER_TOTAL:
                BRSET       RF_IN,PA,START_DATA     ;
                DEC         TMP0
                BNE         SER_TOTAL
                DEC         TMP1
                BNE         SER_TOTAL
                BEQ         NOISE
;*******************************
START_DATA:
                JSR         RF_REC
                BRCLR       RF_RECOK,FLAG,NOISE
                BRSET       STORE_FG,FLAG,STORE_PASS    ;IF MOTOR ON
                                                        ;NO SAVE PASSWORD
                JSR         CHK_PASSWORD
                BRCLR       RF_RECOK,FLAG,NOISE
                BCLR        RF_RECOK,FLAG
ANY_PNT
                BRSET       SEQU_FG,ERROR_FG,RF_SEQU    ;SEUQR ON
                BRCLR       IR_ERR,ERROR_FG,REM_FUNC
                BRCLR       HIGH_LIMIT,LIMIT_SW,DSP_LIGHT
                BCLR        DIR_FG,MOTOR_FG
                JSR         MOTOR_PROC                  ;5/27
DSP_LIGHT:
```


CONFIDENTIAL
LX 1149

```
                JSR         ERROR_LIGHT
                JMP         MAIN
REM_FUNC
                JSR         MOTOR_PROC
NOISE:
                JMP         MAIN
RF_SEQU.
                LDA         #10
                JSR         STORE_LIGHT
                JMP         MAIN
;*********************************
STORE_PASS.
                SEI
                BCLR        RF_RECOK,FLAG
;                BCLR        STORE_FG,FLAG
                LDX         #10
STORE_LOP
                TXA
                JSR         READ_EEP1    ;
                INC         EEP_ADDR
                INC         EEP_ADDR
                JSR         WRITE_EEP    ;
                DECX                     ;
                CPX         #0FFH        ;
                BNE         STORE_LOP    ;
;-----------------------------------------
                CLR         EEP_ADDR
                LDA         REC_DATA0
                STA         EEP_DATA0
                LDA         REC_DATA1
                STA         EEP_DATA1
                JSR         WRITE_EEP    ;
                INC         EEP_ADDR
                LDA         REC_DATA2
                STA         EEP_DATA0
                LDA         REC_DATA3
                STA         EEP_DATA1
                JSR         WRITE_EEP    ;
;*****************************************
                CLR         EEP_ADDR
                JSR         READ_EEP     ;
                LDA         REC_DATA0
                CMP         EEP_DATA0
                BNE         VERY_ERROR
                LDA         REC_DATA1
                CMP         EEP_DATA1
                BNE         VERY_ERROR
```


CONFIDENTIAL
LX 1150

```
                INC         EEP_ADDR
                JSR         READ_EEP    ;

                LDA         REC_DATA2
                CMP         EEP_DATA0
                BNE         VERY_ERROR
                LDA         REC_DATA3
                CMP         EEP_DATA1
                BNE         VERY_ERROR
;                JSR         ON_LIGHT
                LDA         #2
                JSR         STORE_LIGHT
                CLI
                JMP         MAIN
;*****************************************
VERY_ERROR:
                JSR         ERROR_LIGHT
                CLI
                JMP         MAIN
;*******************************
                INCLUDE    "UTIL.ASM"
                INCLUDE    "TIMER.ASM"
                INCLUDE    "EEP.ASM"
                INCLUDE    "LIGHT.ASM"
                INCLUDE    "KEYBOARD.ASM"
                INCLUDE    "AD_COV.ASM"
                INCLUDE    "MOTOR.ASM"
                INCLUDE    "IRQ.ASM"
                INCLUDE    "RF_REC.ASM"
;*******************************
;MOR     WDOG ENABLE
                ORG         1EFFH
                DB          0           ;
                DB          80H
;CLEAR WDOG WRITE 0 TO COPC
                ORG         1FF0H
COPC            DS          1

                ORG         $1FF8
                DW          TIMER_IRQ
                DW          IRQ_VECTOR
                DW          SWI_VECTOR
                DW          RESET

                END
```


CONFIDENTIAL
LX 1151

```
;
;EEPROM MEMORY MAP
;  CODE NO        ADDR        HIGH BYTE     LOW BYTE
;   0             0,1         MSB           LSB
;   1             2,3         MSB           LSB
;   2             4,5         MSB           LSB
;   3             6,7         MSB           LSB
;   4             8,9         MSB           LSB
;   5             A,B         MSB           LSB
;-----------------------------------
;   RANG          12H         UP RANG       DOWN RANG
;
RANG_ADDR         EQU         12H
;*********************************
SIO_STATUS        EQU         0BH
SIO_FG            EQU         7
DATA_COLL         EQU         6
;*********************************
SIO_DATA          EQU         0CH
;*********************************


WREN_CODE:        EQU         %00110000
WRDS_CODE:        EQU         %00000000
ERASE_CODE:       EQU         %11000000
WRITE_CODE:       EQU         %01000000
READ_CODE:        EQU         %10000000

;**************************
;IN DATA ACC
;**************************
SEND_DATA8:
                  STA         EEP_BUF1
                  LDA         #8
                  STA         EEP_CNT

SEND_LOP:
                  ROL         EEP_BUF1
                  BCLR        SDO,PB
                  BCC         SET_DAT0
                  BSET        SDO,PB
SET_DAT0:
                  BSET        SCLK,PB
                  SEC
                  BRSET       SDI,PB,REC_DAT1
                  CLC
```

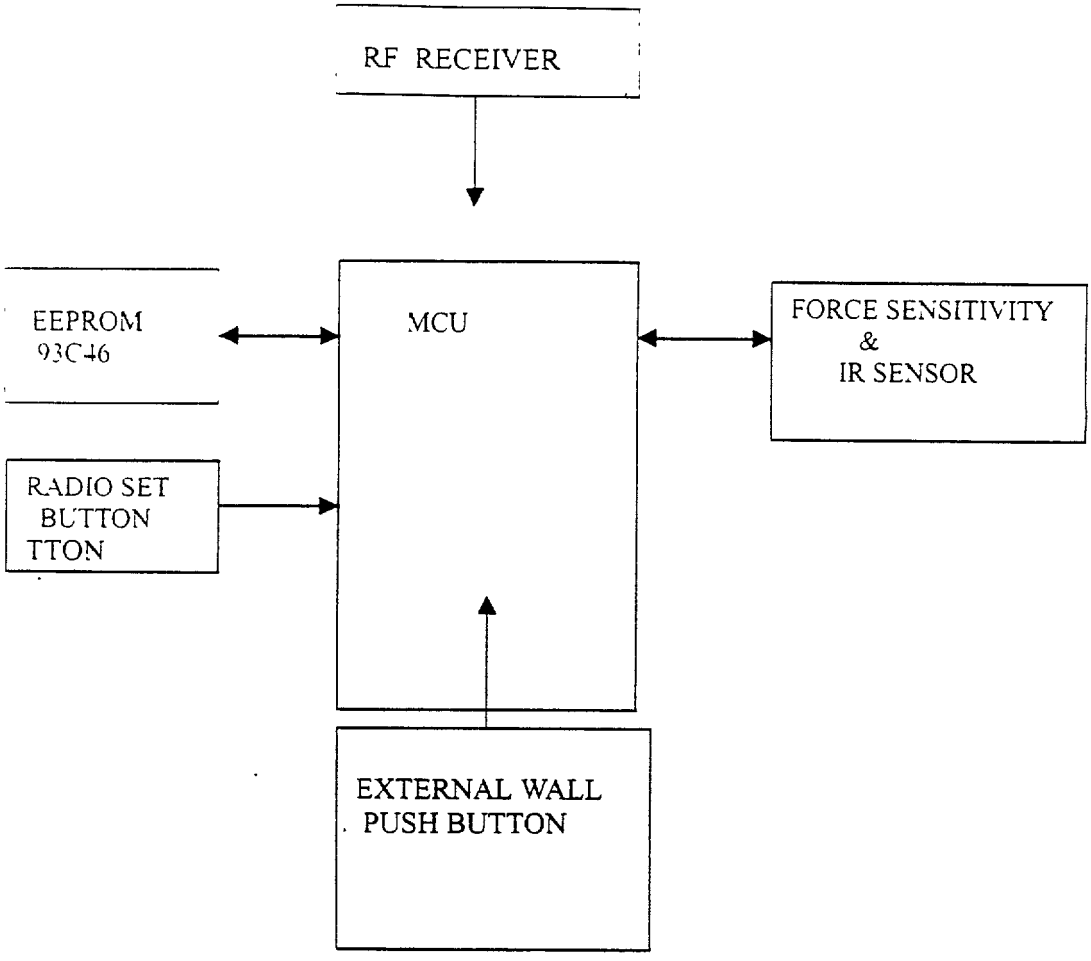

CONFIDENTIAL
LX 1152

```
REC_DAT1
                ROL         EEP_BUF
                BCLR        SCLK,PB

                DEC         EEP_CNT
                BNE         SEND_LOP
                LDA         EEP_BUF
                RTS
;*************************
;READ OUT ACC
;*************************
STAUP_EEP
                BSET        CS93C46,PA
                BRCLR       SDI,PB,$    ;WAIT BUSY
                BCLR        CS93C46,PA
;*****************************
                BSET        CS93C46,PA   ;RE SELECT
                BSET        SDO,PB
                BSET        SCLK,PB
                BCLR        SCLK,PB     ;SEND START
                RTS
;*****************************
READ_EEP1:
                STA         EEP_ADDR
READ_EEP:
                JSR         STAUP_EEP
                LDA         EEP_ADDR
                ORA         #READ_CODE
                JSR         SEND_DATA8
                JSR         SEND_DATA8
                STA         EEP_DATA0
                JSR         SEND_DATA8
                STA         EEP_DATA1
                BCLR        CS93C46,PA   ;RE SELECT
                RTS
;*****************************
WRITE_EEP1:
                STA         EEP_ADDR
WRITE_EEP:
                JSR         STAUP_EEP
                LDA         #WREN_CODE
                JSR         SEND_DATA8
                BCLR        CS93C46,PA   ;WRITE ENABLE

                JSR         STAUP_EEP
                LDA         EEP_ADDR    ;ERASE ADDR
                ORA         #ERASE_CODE
                JSR         SEND_DATA8
```

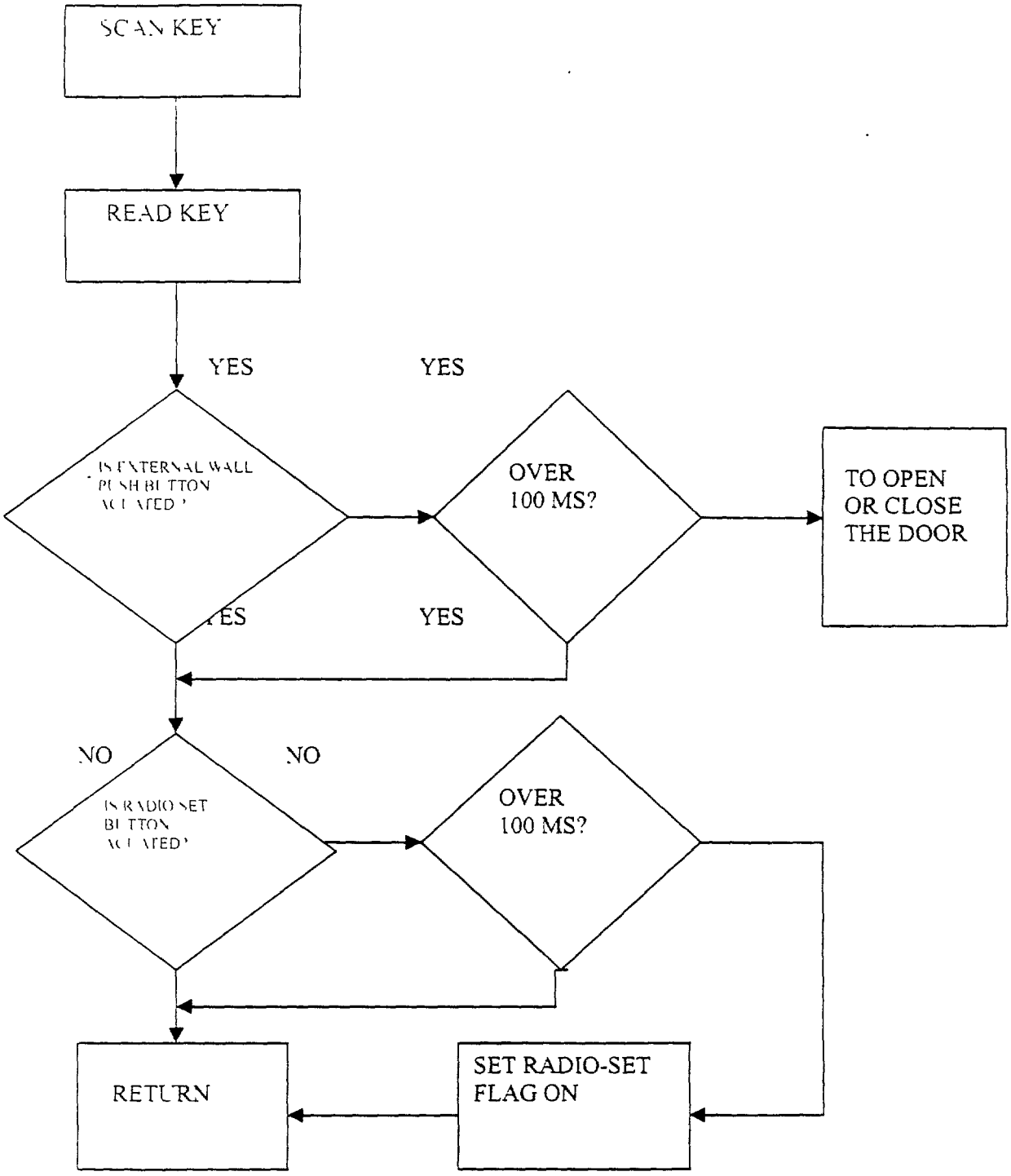

CONFIDENTIAL
LX 1153

```
                BCLR        CS93C46,PA    ;


                JSR         STAUP_EEP
                LDA         EEP_ADDR
                ORA         #WRITE_CODE
                JSR         SEND_DATA8
                LDA         EEP_DATA0
                JSR         SEND_DATA8
                LDA         EEP_DATA1
                JSR         SEND_DATA8
                BCLR        CS93C46,PA    ;

                JSR         STAUP_EEP
                LDA         #WRDS_CODE
                JSR         SEND_DATA8
                BCLR        CS93C46,PA    ;WRITE ENABLE
                RTS
;*************************************
READ_RANG:
                LDA         #RANG_ADDR
                JSR         READ_EEP1
                LDA         #12
                CMP         EEP_DATA1
                LDA         EEP_DATA1
                BCC         SET_DEF_UP
                LDA         #6
SET_DEF_UP:
                STA         UP_RANG
                LDA         #12
                CMP         EEP_DATA1
                LDA         EEP_DATA0
                BCC         SET_DEF_DW
                LDA         #6
SET_DEF_DW.
                STA         DOWN_RANG
                BCS         WRITE_RANG
                RTS
;-------------------------------------
WRITE_RANG.
                LDA         #RANG_ADDR
                STA         EEP_ADDR
                LDA         UP_RANG
                STA         EEP_DATA1
                LDA         DOWN_RANG
                STA         EEP_DATA0
                JMP         WRITE_EEP
;-------------------------------------
```

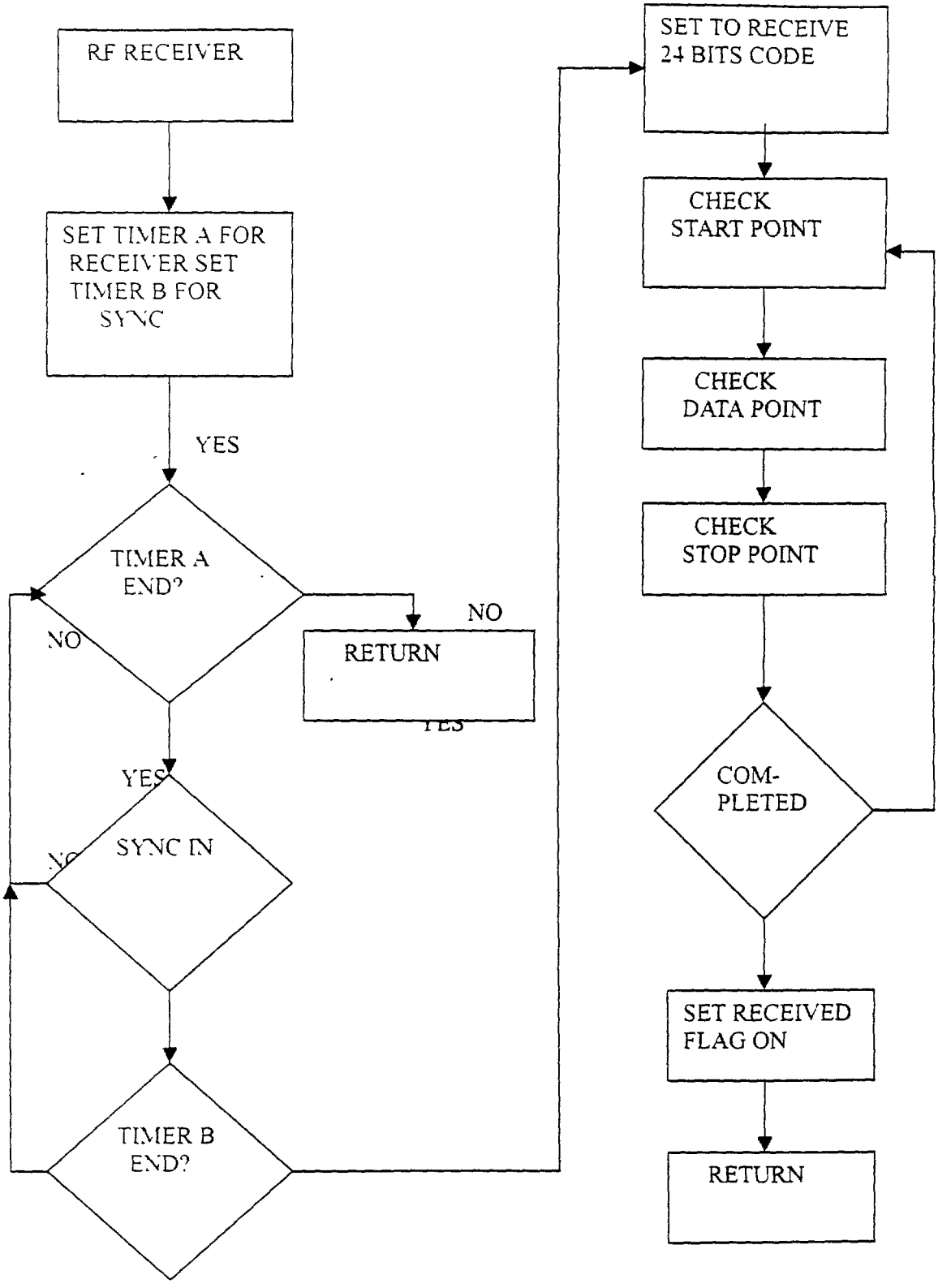

CONFIDENTIAL
LX 1154


RF RECEIVER
EEPROM
93C46
MCU
FORCE SENSITIVITY
&
IR SENSOR
RADIO SET
BUTTON
TTON
EXTERNAL WALL
PUSH BUTTON

EXTERNAL WALL PUSH BUTTON & RADIO SET
BUTTON SCAN KEY CALL INTERRUPT SUB - LOOP



RF RECEIVER CALL INTERRUPT SUB-LOOP



```
IRQ_VECTOR:
                BCLR        IR_ERR,ERROR_FG
                LDA         #IR_TIMEOUT
                STA         IR_CNT
                RTI

SWI_VECTOR:
                RTI
```


CONFIDENTIAL
LX 1158

```
UP_RANG_INC_KEY EQU         0
UP_RANG_DEC_KEY EQU         1
DW_RANG_INC_KEY EQU         2
DW_RANG_DEC_KEY EQU         3
STORE_EEP_KEY   EQU         5             ;4
PB_KEY          EQU         4             ;5
LIGHT_KEY       EQU         6
SEQU_KEY        EQU         7
RANG_MAX        EQU         12
;*******************************

SCANKEY:
;                CLR         ADC_STATUS
                 CLR         PC_DIR
                 CLR         PC
                 NOP
                 LDA         #%01111111
                 AND         PC
                 EOR         #%01111111
;                BSET        6,PC_DIR
;                BSET        6,PC          ;ON BOX LED
                 JSR         ANALY_ADC
                 CMP         KEY_BUFF      ;TEST NEW KEY ?
                 BNE         NEW_KEY
                 LDA         KEY_CNT       ;HV KEY
                 BNE         BMS_NOEND
                 RTS
BMS_NOEND:
                 DEC         KEY_CNT
                 BEQ         KEY_PROC
                 RTS
;*********************************************
NEW_KEY:
                 STA         KEY_BUFF
                 LDA         #50
                 STA         KEY_CNT
                 RTS
SET_LIGHT_ON:
                 JSR         ON_LIGHT
                 RTS
;*************************************
LIGHT_PROC:
                 LDA         LIGHT_CNT0
                 ORA         LIGHT_CNT1
                 BNE         OFF_LIGH
                 LDA         FLAG
                 EOR         #%00001000
```


CONFIDENTIAL
LX 1159

```
                STA         FLAG

                BRSET       LIGHT_ON,FLAG,SET_LIGHT_ON
OFF_LIGH
                BCLR        LIGHT_ON,FLAG
                JMP         CLEAR_LIGHT
;*****************************************
KEY_PROC:
                BRSET       LIGHT_KEY,KEY_BUFF,LIGHT_PROC

                BCLR        SEQU_FG,ERROR_FG
                BRCLR       SEQU_KEY,KEY_BUFF,SEQU_PROC
                BSET        SEQU_FG,ERROR_FG                    ;SET SEQU FG
                BCLR        STORE_FG,FLAG
                BRCLR       MOTOR_ON,MOTOR_FG,SEQU_PROC
                JSR         STOP_MOTOR
SEQU_PROC:
                BRSET       PB_KEY,KEY_BUFF,PBX_PROC
                BRSET       SEQU_KEY,KEY_BUFF,NO_STORES         ;SET NO OP
                BRSET       UP_RANG_INC_KEY,KEY_BUFF,UP_RANG_INC_PRC
                BRSET       UP_RANG_DEC_KEY,KEY_BUFF,UP_RANG_DEC_PRC
                BRSET       DW_RANG_INC_KEY,KEY_BUFF,DW_RANG_INC_PRC
                BRSET       DW_RANG_DEC_KEY,KEY_BUFF,DW_RANG_DEC_PRC
                BRCLR       STORE_EEP_KEY,KEY_BUFF,ASSDOWN_PROC       ;NO K
EY
                JMP         STORE_PROC
ASSDOWN_PROC:
                BRCLR       ASSIGN_DOWN,FLAG,NO_ASSIGN
                BCLR        ASSIGN_DOWN,FLAG
                BRCLR       ASS_DOWN,ERROR_FG,NO_ASS_TIM
                BCLR        ASS_DOWN,ERROR_FG
                JMP         NO_ASSIGN                           ;>2SEC
;*********************************************
NO_ASS_TIM:                                                     ;TIME <2SEC
                BRCLR       HIGH_LIMIT,LIMIT_SW,NO_ASSIGN
                BCLR        DIR_FG,MOTOR_FG
                JSR         MOTOR_PROC
                JSR         ERROR_LIGHT
;*********************************************
NO_ASSIGN.
                BRCLR       EEP_CLRFG,FLAG,NO_STORES
                BCLR        EEP_CLRFG,FLAG
                BSET        STORE_FG,FLAG
                LDA         #60          ;STORE TIME 30SEC
                JSR         STORE_LIGHT
                BCLR        FLASH_FG,FLAG
                BCLR        LIGHT_RELAY,PA
                CLR         CNT0
```


CONFIDENTIAL
LX 1160

```
NO_STORES.
                RTS
;----------------------------------------
UP_RANG_INC_PRC:
                LDX         #UP_RANG
                BNE         RANG_INC_PROC
DW_RANG_INC_PRC:
                LDX         #DOWN_RANG
RANG_INC_PROC:
                LDA         0,X
                CMP         #RANG_MAX
                BCC         RANG__MAX
                INCA
SEC_REP
                STA         0,X
                JSR         WRITE_RANG
                LDA         #250
                STA         KEY_CNT
                JMP         CLEAR_LIGHT
;**************************************
RANG__MAX:
                LDA         #RANG_MAX
                STA         0,X
                JSR         WRITE_RANG
NO_SET_RANG:
                JMP         WARE_LIGHT
;----------------------------------------
UP_RANG_DEC_PRC:
                LDX         #UP_RANG
                BNE         RANG_DEC_PROC
DW_RANG_DEC_PRC:
                LDX         #DOWN_RANG
RANG_DEC_PROC:
                LDA         0,X
                CMP         #0
                BEQ         NO_SET_RANG
                DECA
                JMP         SEC_REP
;**************************************
PBX_PROC:
                BRSET       SEQU_FG,ERROR_FG,SEQU_LIGHT
                BRSET       IR_ERR,ERROR_FG,IR_ERROR_PROC
                JMP         MOTOR_PROC
SEQU_LIGHT:
                LDA         #10
                JMP         STORE_LIGHT
                RTS
;**************************************
```


CONFIDENTIAL
LX 1161

```
IR_ERROR_PROC
                BRCLR       ASSIGN_DOWN,FLAG,NEW_ADOWN ,PUSH BOX 5SEC IN D
OWN
                DEC         ASSIGN_CNT                 ,KEY 50MS
                BNE         ASSIGN_RET
TEST_PNT:
                BRCLR       LOW_LIMIT,LIMIT_SW,CYC_END
                LDA         STP_MTR_CNT
                BNE         CYC_END
                BRSET       MOTOR_ON,MOTOR_FG,CYC_END
                BSET        ASS_DOWN,ERROR_FG
                JSR         CLOSE_DOOR
                JMP         ERROR_LIGHT
CYC_END.
                BCLR        ASSIGN_DOWN,FLAG
                BCLR        ASS_DOWN,ERROR_FG
                RTS
;*************************************
NEW_ADOWN:
                JSR         ERROR_LIGHT
                BSET        ASSIGN_DOWN,FLAG
                LDA         #4
LOAD5SEC:
                STA         ASSIGN_CNT
ASSIGN_RET:
                LDA         #250
                STA         KEY_CNT
NO_STORE:
                RTS
;*****************************
STORE_PROC:
                BRSET       MOTOR_ON,MOTOR_FG,NO_STORE
                BRCLR       EEP_CLRFG,FLAG,NEWCLEAR
                DEC         ASSIGN_CNT
                BNE         ASSIGN_RET
;*****************************
                BCLR        EEP_CLRFG,FLAG
                CLR         EEP_ADDR
                LDA         #0FFH
                STA         EEP_DATA0
                STA         EEP_DATA1

ERASE_LOP.
                JSR         WRITE_EEP
                INC         EEP_ADDR
                LDA         #13
                CMP         EEP_ADDR
                BNE         ERASE_LOP
```


CONFIDENTIAL
LX 1162

```
                LDA         #14
                JMP         STORE_LIGHT
;*****************************
NEWCLEAR
                BSET        EEP_CLRFG,FLAG
                LDA         #10
                BNE         LOAD5SEC
```


CONFIDENTIAL
LX 1163

```
WARE_LIGHT
                LDA         #12
                JMP         STORE_LIGHT
INIT_LIGHT:
                LDA         #10
STORE_LIGHT
                STA         LIGHT_CNT0
                LDA         #0
                STA         LIGHT_CNT1
                BSET        FLASH_FG,FLAG
                BCLR        LIGHT_ON,FLAG
                RTS
;*********************************
CLEAR_LIGHT:
                LDA         #0
                STA         LIGHT_CNT0
                STA         LIGHT_CNT1
                BCLR        LIGHT_ON,FLAG
                RTS
;*********************************
CHK_LIGHT:
                BRSET       LIGHT_ON,FLAG,CONTER_ON
                LDA         LIGHT_CNT0
                ORA         LIGHT_CNT1
                BEQ         LIGHT_OFF    ; LIGHT ON ?

                DEC         LIGHT_CNT0
                BNE         TEST_LIGHT1
                LDA         LIGHT_CNT1
                BEQ         TEST_LIGHT1 ;TIME END ?
                LDA         #120
                STA         LIGHT_CNT0
                DEC         LIGHT_CNT1
TEST_LIGHT1:
                BRSET       FLASH_FG,FLAG,FLASH_PROC
CONTER_ON:
                BSET        LIGHT_RELAY,PA
                RTS
;--------------------------------
FLASH_PROC:
                LDA         #LIGHT_BIT
                EOR         PA
                STA         PA
                RTS
LIGHT_OFF:
                BCLR        LIGHT_RELAY,PA
;************** 99/25/6/
```


CONFIDENTIAL
LX 1164

```
                BRCLR       STORE_FG,FLAG,NO_STORE_LIG
                BRSET       FLASH_FG,FLAG,SET_4M30
                LDA         #6
                JMP         STORE_LIGHT
SET_4M30
                JSR         ON_LIGHT
                BCLR        STORE_FG,FLAG
NO_STORE_LIG
                RTS
;*********************************
ERROR_LIGHT:
SET_FLASH_LIGHT:
                BSET        FLASH_FG,FLAG
                LDA         #0
                JMP         SET_0M30S
ON_LIGHT:
                BCLR       FLASH_FG,FLAG
SET_4M30S.
                LDA         #4
SET_0M30S:
                STA         LIGHT_CNT1
                LDA         #60
                STA         LIGHT_CNT0
                LDA         #250
                STA         CNT0
                BCLR        LIGHT_ON,FLAG     ;
                RTS
```


CONFIDENTIAL
LX 1165

```
;
;
IR_TIMEOUT      EQU     5
MIRROR          EQU     5
;NO CHECK MOTOR
PASS_MTR_TIM    EQU     17
PASS_LIM_IR_TIM EQU     17
PASS_IR_TIM     EQU     17          ;DEFING 5
;*****************************************
STOP_MOTOR:
                LDA     #15
                STA     STP_MTR_CNT         ,STOP 1.7SEC
                BCLR    MOTOR_ON,MOTOR_FG
                BCLR    UP_RELAY,PA
                BCLR    DW_RELAY,PA
                LDA     #41H
                STA     TIMER_CTRL
                CLR     MRT_CNT0
                CLR     MRT_CNT1
                BCLR    ASS_DOWN,ERROR_FG
                BCLR    ASSIGN_DOWN,FLAG
                RTS
;*****************************************
MOTOR_PROC:
                LDA     STP_MTR_CNT
                BEQ     START_MOTOR
                RTS
;*****************************************
START_MOTOR:
                BRSET   MOTOR_ON,MOTOR_FG,STOP_MOTOR
                JSR     READ50HZ
                BRCLR   DIR_FG,MOTOR_FG,OPEN_DOOR
;*************************************
CLOSE_DOOR:
                BRCLR   LOW_LIMIT,LIMIT_SW,OPEN_DOOR
                BCLR    DIR_FG,MOTOR_FG
                JSR     MOTOR_TIME
                BRCLR   MIRROR,PD,MIRR_OPEN
MIRR_CLOSE:
                BSET    MOTOR_ON,MOTOR_FG
                BCLR    UP_RELAY,PA
                BSET    DW_RELAY,PA
                RTS
;*************************************
OPEN_DOOR:
                BRCLR   HIGH_LIMIT,LIMIT_SW,CLOSE_DOOR
                BSET    DIR_FG,MOTOR_FG
                JSR     MOTOR_TIME
```


CONFIDENTIAL
LX 1166

```
                BRCLR       MIRROR,PD,MIRR_CLOSE
MIRR_OPEN:
                BSET        MOTOR_ON,MOTOR_FG
                BCLR        DW_RELAY,PA
                BSET        UP_RELAY,PA
                RTS
;*****************************************
CHK_MOTOR:
                DEC         IR_CNT
                BNE         CHK_MOTR_SPEED
SET_IR_ERR.
                BIL         NO_IR                   ;IS IRQ IS LOW NO SET FL
AG
                BSET        IR_ERR,ERROR_FG         ;
                LDA         #IR_TIMEOUT
                STA         IR_CNT                  ;RE SAMPLE 8MS
                BRCLR       MOTOR_ON,MOTOR_FG,CHK_MOTR_SPEED
                BRSET       DIR_FG,MOTOR_FG,CHK_MOTR_SPEED          ;DOWN NO
 CHECK IR
                BRSET       ASS_DOWN,ERROR_FG,CHK_MOTR_SPEED
                LDA         PASS_IR                 ;PASS TIME OUT
                BNE         CHK_MOTR_SPEED
                JSR         STOP_MOTOR
TEST_POINT.
                                                    ;IR ERROR
                LDA         #8
                JSR         STORE_LIGHT
                BSET        IR_ERROR,MOTOR_FG
                JMP         CHK_MOTR_SPEED

NO_IR:
                BCLR        IR_ERR,ERROR_FG
;*****************************
CHK_MOTR_SPEED:                         ;CHECK FLAG
                LDA         PASS_MTR
                BNE         CHK_LIMIT_SW
                BRCLR       RPM_ERR,ERROR_FG,CHK_LIMIT_SW
                BRCLR       MOTOR_ON,MOTOR_FG,CHK_LIMIT_SW
                JSR         STOP_MOTOR
                LDA         #8
                JSR         STORE_LIGHT
                BSET        LOW_SPEED,MOTOR_FG
;*********************************
CHK_LIMIT_SW.
                LDA         #%00110000
                AND         PA
                CMP         LIMIT_SW
                BNE         NEW_LIM_TIM ;NEW LIMIT SW
```


CONFIDENTIAL
LX 1167

```
                LDA         LIMIT_CNT
                BEQ         CHK_MTR_2MS     ;BMS TIME 0 1 SEC
                DEC         LIMIT_CNT
                BNE         CHK_MTR_2MS     ;BMS TIME 0 1 SEC
;************************************
                BRCLR       MOTOR_ON,MOTOR_FG,CHK_MTR_2MS   ,MOTOR OFF

                BRSET       DIR_FG,MOTOR_FG,CHK_UP_LIM      ,MOTOR ON
                BRCLR       MIRROR,PD,MIRR_LIM_DW           ;MIRROR MODE
MIRR_LIM_UP:
                BRSET       LOW_LIMIT,LIMIT_SW,CHK_MTR_2MS
                JSR         STOP_MOTOR
                RTS
CHK_UP_LIM:
                BRCLR       MIRROR,PD,MIRR_LIM_UP
MIRR_LIM_DW:
                BRSET       HIGH_LIMIT,LIMIT_SW,CHK_MTR_2MS
                JSR         STOP_MOTOR
                RTS
;*************************
NEW_LIM_TIM:
                STA         LIMIT_SW
;               LDA         #50
                LDA         #8
                STA         LIMIT_CNT
CHK_MTR_2MS:

                DEC         MOTOR_CNT0
                BEQ         MOTOR01S
                RTS
;*************************
MOTOR01S:
                LDA         #50
                STA         MOTOR_CNT0
                BRSET       MOTOR_ON,MOTOR_FG,MOTOR_ISON
                BCLR        UP_RELAY,PA
                BCLR        DW_RELAY,PA
                LDA         STP_MTR_CNT
                BEQ         STP_WAIT_OK
TEST_STP:
                DEC         STP_MTR_CNT
;                LDA         #10
;                CMP         STP_MTR_CNT
;                BCS         HOLD_RELAY
RELS_RELAY:
;                BCLR        MOTOR_DIR,PA
HOLD_RELAY:
                RTS
```


CONFIDENTIAL
LX 1168

```
STP_WAIT_OK
                BRSET       MRT_ERR,ERROR_FG,REV_DOOR    ;
                BRSET       LOW_SPEED,MOTOR_FG,REV_DOOR    ;
                BRSET       IR_ERROR,MOTOR_FG,REV_DOOR    ;
                JMP         RELS_RELAY
REV_DOOR
                BRSET       DIR_FG,MOTOR_FG,CLEAR_SPEED
                BRCLR       MIRROR,PD,MIRR_SAFE_UP
                BRCLR       HIGH_LIMIT,LIMIT_SW,CLEAR_SPEED
                BRSET       HIGH_LIMIT,LIMIT_SW,SAFE_UP
MIRR_SAFE_UP:
                BRCLR       LOW_LIMIT,LIMIT_SW,CLEAR_SPEED
SAFE_UP
                JSR         MOTOR_PROC                          ;MOTOR SPEED
                JSR         ERROR_LIGHT                         ;ERROR DOWN
CLEAR_SPEED.                                                    ;PROC
                BCLR        LOW_SPEED,MOTOR_FG
                BCLR        IR_ERROR,MOTOR_FG
                BCLR        MRT_ERR,ERROR_FG
MTR_STP_OK:
                RTS
;*****************************************
MOTOR_ISON:
;                LDA         WAIT_START
;                BEQ         POWER_MOTOR
;                DEC         WAIT_START      ;WAIT RELAY OK
;                RTS
POWER_MOTOR:
;                BSET        MOTOR_RELAY,PA
                LDA         PASS_MTR
                BEQ         CHK_IR_TIM      ;CHK  MOTOR RPM
                DEC         PASS_MTR
CHK_IR_TIM:
                LDA         PASS_IR
                BEQ         CHK_MRT_TIM
                DEC         PASS_IR
CHK_MRT_TIM:
                DEC         MRT_CNT0
                BEQ         MRT_TIMEND
                RTS
MRT_TIMEND
                LDA         MRT_CNT1
                BEQ         SET_MRT_ERROR
                DEC         MRT_CNT1
                RTS
SET_MRT_ERROR:
                BSET        MRT_ERR,ERROR_FG
                JSR         STOP_MOTOR
```


CONFIDENTIAL
LX 1169

```
                JMP         SET_FLASH_LIGHT

;*****************************************

MOTOR_TIME
                LDA         #200
                STA         MRT_CNT0     ;SET 20SEC
                CLR         MRT_CNT1
                BRCLR       MRT_SEL,PA,STORE_MRT
                LDA         #280-256
                STA         MRT_CNT0
                LDA         #1
                STA         MRT_CNT1     ;SET 28SEC
STORE_MRT
                LDA         #50
                STA         MOTOR_CNT0   ;0.1SEC
;*****************************************
;LOAD PASS TIME
;*****************************************
                LDA         #PASS_MTR_TIM
                STA         PASS_MTR
                BCLR        LOW_SPEED,MOTOR_FG
                LDA         #PASS_LIM_IR_TIM    ;1.7SEC
                BRCLR       HIGH_LIMIT,PA,PASS_TIM_OK
                BRCLR       LOW_LIMIT,PA,PASS_TIM_OK
                LDA         #PASS_IR_TIM        ;0.5SEC
PASS_TIM_OK:
                STA         PASS_IR
                JSR         ON_LIGHT
                JSR         READ_RANG
                JSR         SET_READ_RPM
                RTS

;*****************************************
READ50HZ:                                   ;SET 1= 50HZ
                BCLR        SCLK,PB_DIR     ;CHANG TO IN
                BSET        HZ_50,MOTOR_FG
                NOP
                NOP
                BRSET       SCLK,PB,SET50HZ     ;TEST
                BCLR        HZ_50,MOTOR_FG
SET50HZ.
                BSET        SCLK,PB_DIR     ;CHANG TO OUT
                RTS
```


CONFIDENTIAL
LX 1170

```
RF_INPUT        EQU     7

RF_REC
                BRCLR   RF_INPUT,PA,NO_RFIN
                BSET    CS93C46,PA
                BRCLR   RF_INPUT,PA,NO_RFIN     ;START BIT

                CLR     REC_DATA0
                CLR     REC_DATA1
                CLR     REC_DATA2
                CLR     REC_DATA3
                LDA     #28
                STA     REC_CNT

REC_LOP:                                        ;
                LDA     80                      ;
                STA     TMP0                    ;
SER_STARTBIT:
                BRCLR   RF_INPUT,PA,SER_START_OK ;SEAR START BIT
                DEC     TMP0                    ;
                BNE     SER_STARTBIT            ;
                BEQ     NO_RFIN                 ;     DATA STOP
SER_START_OK:                                   ;     _________
                JSR     DLY_250US               ;   ^
                BRCLR   RF_INPUT,PA,WAIT_DATA   ; 250
                NOP                             ;
                BRSET   RF_INPUT,PA,NO_RFIN     ; us
                                                ;
WAIT_DATA:                                      ;       ^
                JSR     DLY_250US               ;     READ DATA BIT
                JSR     DLY_250US               ;     READ DATA BIT
                CLC
                BRCLR   RF_INPUT,PA,SET_DATA0   ; 120
                SEC
SET_DATA0:
                ROL     REC_DATA0
                ROL     REC_DATA1
                ROL     REC_DATA2
                ROL     REC_DATA3

                JSR     DLY_250US               ;     WAIT STOP
                LDA     #80H
                STA     TMP0
WAIT_STOPBIT:
                BRSET   RF_INPUT,PA,STOPBIT     ; 120
                DEC     TMP0                    ;
```


CONFIDENTIAL
LX 1171

```
                BNE         WAIT_STOPBIT                ;
                BEQ         NO_RFIN                     ; FAIL
STOPBIT
                JSR         DLY_250US                   ;
                BRCLR       RF_INPUT,PA,NO_RFIN         ; TEST STOPBIT
                DEC         REC_CNT
                BNE         REC_LOP
;-----------------------------------------
REC_TEST.
                LDA         REC_DATA0
                ORA         REC_DATA1
                ORA         REC_DATA2
                ORA         REC_DATA3
                BEQ         RF_RELS_OK
                LDA         #0FFH
                CMP         REC_DATA0
                BNE         RECOK
                CMP         REC_DATA1
                BNE         RECOK
                CMP         REC_DATA2
                BNE         RECOK
                CMP         REC_DATA3
                BEQ         RF_RELS_OK
;-----------------------------------
RECOK:
                LDA         RF_OLD_CNT
                BNE         RF_RELS_OK
                BSET        RF_RECOK,FLAG
RF_RELS_OK:
                LDA         #7
                STA         RF_OLD_CNT
                LDA         #DLY01S
                STA         RF_OLD_CNT1
NO_RFIN:
                RTS
;*******************************
DLY_250US:
                LDA         #52         ;52
                STA         TMP0
DLY120LOP:
                DEC         TMP0
                BNE         DLY120LOP
                RTS
;*******************************
DLY_100US:
                STA         ADC_TMP1
DLY100LOP:
                DEC         ADC_TMP1
```


CONFIDENTIAL
LX 1172

```
                BNE         DLY100LOP
                RTS
;*******************************
CHK_PASSWORD
                LDA         #0
                STA         EEP_ADDR
CHK_LOOP
                JSR         READ_EEP
                INC         EEP_ADDR
                LDA         EEP_DATA0
                CMP         REC_DATA0
                BNE         NO_MACHWORD
                LDA         EEP_DATA1
                CMP         REC_DATA1
                BNE         NO_MACHWORD
                JSR         READ_EEP
                LDA         EEP_DATA0
                CMP         REC_DATA2
                BNE         NO_MACHWORD
                LDA         EEP_DATA1
                CMP         REC_DATA3
                BEQ         PASSWORD_OK
;*************************************
NO_MACHWORD:
                INC         EEP_ADDR
                LDA         #11
                CMP         EEP_ADDR
                BCC         CHK_LOOP
                BCLR        RF_RECOK,FLAG

PASSWORD_OK:
                RTS
;*************************************
```


CONFIDENTIAL
LX 1173

```
TIMER_CTRL      EQU     12H
IN_CAP_IRQ      EQU     7
OUT_CMP_IRQ     EQU     6
TIME_OVER_IRQ   EQU     5
INPUT_EDGE      EQU     1
OLVL            EQU     0
;*********************************
TIMER_STATUS    EQU     13H
IN_CAP_FG       EQU     7
OUT_CMP_FG      EQU     6
TIMER_OVER_FG   EQU     5
;*********************************
TIMER_H         EQU     18H
TIMER_L         EQU     19H
ATIMER_H        EQU     1AH
ATIMER_L        EQU     1BH
IN_CAP_H        EQU     14H
IN_CAP_L        EQU     15H
OUT_COMP_H      EQU     16H
OUT_COMP_L      EQU     17H
;*********************************
SET_READ_RPM:
                SEI
                LDA     TIMER_STATUS
                LDA     TIMER_L
                LDA     IN_CAP_H
                STA     RPM_CNT1
                LDA     IN_CAP_L
                STA     RPM_CNT0
                CLR     RPM_CNT2
                LDA     TIMER_L
                LDA     #%11000000    ;SET COMP ,INCAP  ,OVER
                STA     TIMER_CTRL
                LDA     #15
                STA     SENSOR_CNT
                BCLR    SAMP_OK,MOTOR_FG
                CLI
                RTS
;*************************************
INIT_TIME:
;               SEI
                LDA     #3
                STA     OUT_COMP_H
                LDA     #0FCH
                STA     OUT_COMP_L
                LDA     TIMER_STATUS        ;CLEAR FLAG
```


CONFIDENTIAL
LX 1174

```
                LDA         TIMER_L
                LDA         OUT_COMP_L
                LDA         IN_CAP_L
                LDA         #%01000001
                STA         TIMER_CTRL
                LDA         #250
                STA         CNT0
                RTS
;*********************************
TIMER_IRQ:
                BRCLR       IN_CAP_IRQ,TIMER_CTRL,TIMIRQ_RET
                BRCLR       IN_CAP_FG,TIMER_STATUS,NO_INCAP_IRQ1
                JSR         INCAP_IRQ
NO_INCAP_IRQ1:
                BRCLR       TIMER_OVER_FG,TIMER_STATUS,TIMIRQ_RET
OVER_IRQ
                INC         RPM_CNT2
                LDA         TIMER_L
                LDA         #2
                CMP         RPM_CNT2
                BCC         TIMIRQ_RET
OVER_TEST:
                BSET        RPM_ERR,ERROR_FG
                CLR         RPM_CNT2
TIMIRQ_RET.
                TST         RF_OLD_CNT              ;-----
                BEQ         CHK_2MSA                ;2000/1/13 INSTER
                DEC         RF_OLD_CNT1             ;
                BNE         CHK_2MSA                ;
                LDA         #DLY01S                 ;
                STA         RF_OLD_CNT1             ;
                DEC         RF_OLD_CNT              ;
CHK_2MSA:                                           ;---
                BRCLR       OUT_CMP_FG,TIMER_STATUS,NO_COMP_IRQ
                JSR         COMP_IRQ
NO_COMP_IRQ:
                RTI
;*********************************
COMP_IRQ:
                LDA         OUT_COMP_H
                STA         TIM_TMP1
                LDA         OUT_COMP_L

                ADD         #256-24
                STA         TIM_TMP0
                LDA         #3
                ADC         TIM_TMP1
```


CONFIDENTIAL
LX 1175

```
                STA         OUT_COMP_H
                LDA         TIM_TMP0
                STA         OUT_COMP_L
;                LDA         RF_OLD_CNT
;                BEQ         RF_BMS
;                DEC         RF_OLD_CNT
;RF_BMS:
                JSR         CONTER_2MS
                JSR         SCANKEY
                BSET        SAMP_AD,ERROR_FG
NO_OVER.
                RTS
;*********************************************
CONTER_2MS.
;                TST         RF_OLD_CNT
;                BEQ         CHK_2MSA                ;2000/1/13
;                DEC         RF_OLD_CNT1
;                BNE         CHK_2MSA
;                LDA         #DLY01S
;                STA         RF_OLD_CNT1
;                DEC         RF_OLD_CNT
;CHK_2MSA:
                JSR         CHK_MOTOR
                DEC         CNT0
                BNE         NO_OVER05SEC
                LDA         #250
                STA         CNT0
                JSR         CHK_LIGHT
NO_OVER05SEC:
                RTS
;*************************************
INCAP_IRQ.
                DEC         SENSOR_CNT
                BEQ         COMP_RPM
                LDA         IN_CAP_L    ;CLEAR IRQ
                RTS
;*************************************
COMP_RPM:
                LDA         IN_CAP_H    ;
                STA         RPM_TMP1
                LDA         IN_CAP_L    ;
                STA         RPM_TMP0
                CLR         RPM_CNT2
                BSET        SAMP_OK,MOTOR_FG
                LDA         #15
                STA         SENSOR_CNT

                LDA         RPM_TMP0
```


CONFIDENTIAL
LX 1176

```
                SUB         RPM_CNT0
                STA         RPM_CAL0

                LDA         RPM_TMP1
                SBC         RPM_CNT1
                STA         RPM_CAL1     ;CAL TIMER


                LDA         RPM_TMP0
                STA         RPM_CNT0
                LDA         RPM_TMP1
                STA         RPM_CNT1     ,UPDATE


                RTS
;***********************************************

TEST_RPMS:
                BRCLR       SAMP_OK,MOTOR_FG,RPM_RET
                BCLR        SAMP_OK,MOTOR_FG

                                         ;COMP RPM
                BRSET       DIR_FG,MOTOR_FG,UP_SPEED
                LDA         DOWN_RANG
                CLC
                ROLA
                TAX
;---------------------------------------
                BRSET       HZ_50,MOTOR_FG,DW_READ_50TAB
                LDA         DW_60TAB,X
                STA         RTMP1
                INCX
                LDA         DW_60TAB,X
                STA         RTMP0
                JMP         CAL_LOOP
DW_READ_50TAB:
                LDA         DW_50TAB,X
                STA         RTMP1
                INCX
                LDA         DW_50TAB,X
                STA         RTMP0
                JMP         CAL_LOOP
;***************************************************
UP_SPEED:
                LDA         UP_RANG
                CLC
                ROLA
                TAX
```


CONFIDENTIAL
LX 1177

```
                BRSET       HZ_50,MOTOR_FG,UP_READ_50TAB
                LDA         UP_60TAB,X
                STA         RTMP1
                INCX
                LDA         UP_60TAB,X
                STA         RTMP0
                JMP         CAL_LOOP
UP_READ_50TAB.
                LDA         UP_50TAB,X
                STA         RTMP1
                INCX
                LDA         UP_50TAB,X
                STA         RTMP0
                JMP         CAL_LOOP
CAL_LOOP:
                LDA         RTMP0
                SUB         RPM_CAL0
                LDA         RTMP1
                SBC         RPM_CAL1
                BCS         RPM_ERROR
                BCLR        RPM_ERR,ERROR_FG
RPM_RET:
                RTS

RPM_ERROR:
                CLR         RPM_CNT2
                BSET        RPM_ERR,ERROR_FG
                RTS

;***********************************************

UP_50TAB:
                DW          21111       ;0 ;1500
                DW          21111+(72*2)        ;1
                DW          21111+(72*4)        ;2
                DW          21111+(72*6)        ;3
                DW          21111+(72*8)        ;4
                DW          21111+(72*10)        ;5
                DW          21111+(72*12)        ;6
                DW          21111+(72*14)        ;7
                DW          21111+(72*16)        ,8
                DW          21111+(72*18)        ;9
                DW          21111+(72*20)        ;0
                DW          21111+(72*22)        ;1
                DW          21111+(72*24)        ;2
DW_50TAB:
                DW          20904       ;1500
```


CONFIDENTIAL
LX 1178

```
                DW          20904+(72*2)        ,1
                DW          20904+(72*4)   ;2
                DW          20904+(72*6)    ;3
                DW          20904+(72*8)    ,4
                DW          20904+(72*10)   ;5
                DW          20904+(72*12)   ;6
                DW          20904+(72*14)   ,7
                DW          20904+(72*16)   ;8
                DW          20904+(72*18)   ;9
                DW          20904+(72*20)   ;0
                DW          20904+(72*22)   ;1
                DW          20904+(72*24)   ;2
;*******************************
UP_60TAB:
                DW          17590        ;0
                DW          17590+(60*2) ;1
                DW          17590+(60*4) ;2
                DW          17590+(60*6) ;3
                DW          17590+(60*8) ;4
                DW          17590+(60*10) ;5
                DW          17590+(60*12) ;6
                DW          17590+(60*14) ;7
                DW          17590+(60*16) ;8
                DW          17590+(60*18)  ;9
                DW          17590+(60*20) ;10
                DW          17590+(60*22) ;11
                DW          17590+(60*24) ;12
;*******************************
DW_60TAB:
                DW          17292         ;0 16.5kg 1setp 2.5kg
                DW          17292+(22*4)      ;1 19kg
                DW          17292+(22*8) ;2 21.5
                DW          17292+(22*12) ;3
                DW          17292+(22*16) ;4
                DW          17292+(22*20) ;5
                DW          17292+(22*24) ;6
                DW          17292+(22*28) ;7
                DW          17292+(22*32) ;8          18872
                DW          17292+(22*36) ;9            ;
                DW          17292+(22*40) ;0    16952+40*n
                DW          17292+(22*44) ;1
                DW          17292+(22*48) ;2 18392   ;19
                                  ;50kg 33
```


CONFIDENTIAL
LX 1179

```
CLEAR_RAM:
                LDX         #50H
CLR_LOP:
                CLR         0,X
                INCX
                CPX         #0C0H
                BNE         CLR_LOP
                RTS
;*********************************
DELAY_1S
                CLR         TMP0
                CLR         TMP1

DELAY_LOP:
                NOP
                NOP
                NOP
                DEC         TMP0
                BNE         DELAY_LOP
                LDA         #0
                STA         COPC
                DEC         TMP1
                BNE         DELAY_LOP
                RTS
;*********************************
INIT_IO:
                LDA         #0
                STA         PA
                LDA         #%00001111
                STA         PA_DIR

                LDA         #0
                STA         PB
                LDA         #%10111111
                STA         PB_DIR
                LDA         #0h
                STA         PC_DIR      ;ALL IN
                LDA         #%11011111
                STA         PD_DIR      ;ALL OUT
                RTS

;*********************************
```


CONFIDENTIAL
LX 1180

REISSUE LITIGATION

ATTORNEY DOCKET NO. 59838

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicants:	Carl Heitschel Colin Willmott Wayne Schindler
Appln. No.:	08/700,610
Filed:	August 12, 1996
Title:	CODING SYSTEM FOR MULTIPLE TRANSMITTERS AND A SINGLE RECEIVER FOR A GARAGE DOOR OPENER
Group Art Unit:	2745
Examiner:	Edwin C. Holloway III

Received
JUN 23 1998
Group 2700

DECLARATION OF JAMES DAVID ROLLS

James David Rolls declares that:

1. I am Executive Vice President, Sales and Marketing of The Chamberlain Group, Inc., the assignee of the aforementioned application, hereafter "Chamberlain." I became employed by Chamberlain in 1983 as a Vice President of Sales and Marketing for the trade side of the business. I assumed my present position in 1993.

2. In 1992 Chamberlain changed over its entire garage door operator line to operators that are electronically programmable for two or more different RF transmitters having unique fixed, non-user changeable codes.

3. Under my direction IMR, a marketing research firm made annual estimates of the market shares of major brands of

garage door operators. Based upon that market research I estimated the market shares of major brand garage door operators as follows.

4. In 1993 Chamberlain had about 57% of the market for major brands of garage door operators. Its total sales of garage door operators was about $206,288,000.

5. In 1994[1] Chamberlain had about 60% of the market for major brands of garage door operators. Its total sales of garage door operators was about $225,540,000.

6. In 1995 Chamberlain had about 62% of the market for major brands of garage door operators. Its total sales of garage door operators was about $220,265,000.

7. In 1996 Overhead Door, one of Chamberlain's major competitors, changed over its product line to garage door operators having electronic code programming for two or more different RF transmitters.

8. In 1996 Chamberlain had about 62% of the market for major brands of garage door operators. Overhead Door's share of the market for major brand garage door operators that was about 24%. I estimate the total market share of major brand garage door operators attributable to Chamberlain's electronic code technology to be about 86%. Chamberlain's total sales of garage door operators was about $242,744,000.

9. In 1997 Chamberlain had about 58% of the market for major brands of garage door operators. Overhead Door's share of the market for major brands of garage door operators was about 33%. I estimate the total market share of major brand garage door operators attributable to Chamberlain's electronic

---

[1] A portion of the garage door operators sold by Chamberlain in 1993 and 1994 had application specific integrated circuits rather than microprocessors.

code technology to be about 91%. Chamberlain's total sales of garage door operators was about $272,480,000.

10. The total sales of Chamberlain's garage door operators from 1993 through 1997 was about $1,167,317,000.

All statements made of my own knowledge are true, and all statements made on information and belief are believed to be true. I have been warned that willful, false statements and the like are punishable by fine or imprisonment, or both (18 U.S.C. § 1001) and may jeopardize the validity of the application or of any patent issuing thereon.

James David Rolls

June 19, 1998

ATTORNEY DOCKET NO. 59838

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicants:	Carl Heitschel Colin Willmott Wayne Schindler	)
Appln. No.:	08/700,610	)
Filed:	August 12, 1996	)
Title:	CODING SYSTEM FOR MULTIPLE TRANSMITTERS AND A SINGLE RECEIVER FOR A GARAGE DOOR OPENER	)
Group Art Unit	2745	)
Examiner:	Edwin C. Holloway III	)

Received
OCT 23 1998
Group 2700

DECLARATION OF V. THOMAS RHYNE

Dr V. Thomas Rhyne declares that:

1. I have been retained by The Chamberlain Group, Inc., the assignee of the aforementioned application.

2 I have been teaching and practicing the art of electrical engineering for over thirty-five years, including extensive experience with both analog and digital electronics. I hold degrees from Mississippi State University (BSEE, 1962), the University of Virginia (MEE, 1964), and the Georgia Institute of Technology (Ph.D. in EE, 1967). I am a registered Professional Engineer in the State of Texas. I taught digital systems design and computer science at Texas A&M University from 1967 to 1983, and computer engineering at the University of Texas from 1983 to 1990. My industrial experience includes work for Texas Instruments, Control Data

Corporation, NASA, Texas Digital Systems, Inc., a company I co-founded to produce microprocessor-based computer peripherals in 1976, the Micro-electronics and Computer Technology Corporation (MCC), and Motorola, Inc

3. I am an experienced programmer with proficiency in assembly programming on a wide variety of microprocessors and in a number of high-level languages. I also have experience with wireless transmitters and receivers and with digital coding. My experience and qualifications have been recognized by the Texas Society of Professional Engineers (Young Engineer of the Year in Texas, 1973), the American Society for Engineering Education (Terman Awardee, 1980), the Institute of Electrical and Electronics Engineers (IEEE Fellow, 1990), and the Accreditation Board for Engineering and Technology (ABET Fellow, 1992) I am the author of over thirty technical papers and an award winning textbook, Fundamentals of Digital System Design, published by Prentice-Hall in 1973 and adopted at over thirty U.S. and international universities.

4 I have worked part-time as a consulting engineer for the past twenty-five years.

5 My curriculum vitae is attached as Exhibit A

6 I have read the specification of application serial no 700,610, (hereinafter the '610 application), the Office Actions and the amendment in the prosecution history of that application, and the prior art that was cited during the

LX 1483

prosecution of the application which issued as patent no. Re 35,364.

7. It is my opinion that the level of ordinary skill in the art of the '610 application is that of an individual who, as of October 1985, had a college degree in electrical or computer engineering and at least two years of experience in designing microprocessor based control systems, including significant experience with the creation of microprocessor software programs for controlling such systems and with wireless garage door opener (GDO) systems.

8. I have carefully reviewed the specification of the '610 application, finding that the '610 specification clearly discloses to one skilled in the art applicable to that application the description of two embodiments for storing and locating transmitter identity codes in the memory of the receiver.

9. As the Examiner has recognized, the first of these embodiments includes a system incorporating a GDO which makes use of a hardware memory selection switch 23 for determining the specific memory location at which each newly received transmitter identity code is stored as it is programmed (or learned) by the GDO Because the disclosed memory selection switch 23 is of a hardware nature, a manual action by the user is required to move the switch from a first position to a second position, and so forth, as successive transmitter identity codes are programmed into the opener.

10. In addition, however, my study of the '610 application shows that it also provides a clear, obvious, and readily understandable disclosure of another embodiment of the disclosed GDO system. This second embodiment includes a software memory selection switch, referred to in the specification and drawings of the '610 application as the "code location pointer." The actions of this software memory selection switch are fully described in Figures 3 and 4 of the '610 application as well as at Column 4, Lines 31 through 68 of the specification. Unlike the hardware version of the memory selection switch, the software memory selection switch "moves" from a first position to a second position and so forth, automatically, returning to the starting memory position only when all available positions have been used to store programmed transmitter identity codes. No form of user selectability is provided (or allowed) for the software memory location switch.

11. As stated above, the '610 application refers to the software memory selection switch as a "code location pointer " This term, in and of itself, clearly designates a software version of the memory selection switch. "Code" refers to the new transmitter identity code that is to be programmed into the GDO. "Location" refers to the specific memory address at which that identity code is to be stored and is a term commonly used to designate such addresses as shown, for example, at Column 4 Lines 37 to 39 of the specification The

term "pointer" is (and was, as of October 1985) a commonly used and well understood term for a stored value which identifies the memory address of another value.

12. The IEEE Standard Dictionary of Electrical and Electronics Terms, Third Edition, (The Institute of Electrical and Electronic Engineers, Inc.) (1984) (Exhibit B), for example, defines "pointer" as "a data item whose value is the location of another data item," and the widely adopted textbook, Introduction to Microprocessors: Software, Hardware, Programming, by Lance Leventhal, (Prentice-Hall) (1978) (Exhibit C) defines "pointer" as "[a] register or memory location that contains an address rather than data." See also The 8086 Family User Manual, (Intel Corporation) (1980) at 2-35 (Exhibit D); Microprocessor Software, Max J. Schindler, Ed., (1980) at pp. 65, 67 (Exhibit E).

13. In the case of the "code location pointer", its value, as disclosed in the '610 application, serves to identify the location within the memory of the GDO system at which the next transmitter identity code to be learned is to be stored. This is precisely what the term "pointer" connotes in the computer science art.

14. Figures 3 and 4 of the '610 application describe, in a form familiar to those skill in the art as of October 1985, the software-based control operations of the microprocessor of the GDO system. In the '610 application, that form is referred to as a flow diagram, a term and a graphical format

that are both well understood by those of skill in the art as describing the operations of a software program. More specifically, a flow diagram or chart describes in a graphical format the steps executed by a stored program machine, e.g., a microprocessor unit. The rectangles in Figures 3 and 4 define specific actions which are performed by the system under software control, and the diamonds define test points at which a branch may be made based on the results of the test defined within the diamond. An example of this type of description is shown in Introduction to Microcomputers & Microprocessors, by Arpad Barna and Dan I. Porat, (John Wiley & Sons, Inc.) (1976) at p. 18, (Exhibit F).

15. The two lower right blocks of Figure 3 describe the operation of the portion of the disclosed software control program executing in the microprocessor unit 44 which uses the code location pointer as the software memory selection switch to determine the location in the GDO system's memory at which each newly learned transmitter identity code is to be stored As one of skill in the art would readily recognize, these two blocks define, in typical and well understood flow chart form, a sequence of software operations executed by the microprocessor unit 44 and referred to in the art of computer science as a "loop". A loop is a set of microprocessor operations which is repeated until an end-of-loop condition is reached

16 References describing the construction and use of software loops exactly as disclosed for controlling the code location pointer were readily available as of October 1985. The previously cited IEEE Dictionary at p. 503 (Exhibit B), for example, defines "loop" as "a sequence of instructions that is executed repeatedly until a terminal condition prevails," and the Leventhal textbook defines "loop" as "a self-contained sequence of instructions that a processor repeats until a terminal condition is reached. A conditional jump instruction can determine if the loop should be continued of terminated." Leventhal, in fact, devotes a twenty-page section of his textbook to the subject of "Loops and Arrays" (Section 5.2, pages 179 to 198) (Exhibit C).

17. A loop includes a body containing the instructions which are the principal business of the loop. An adjustment step is embodied in a set of instructions that adds to a counter of the number of repetitions already performed. An exit from the loop is provided by way of a conditional branch. The conditional branch involves a test for occurrence of certain conditions, as shown, for example, in Assembler Language Programming: The IBM System 360, by George Struble, (Addison-Wesley Publishing Co.) (1971) at pp. 148-151 (Exhibit G)

18 As the specification of the '610 application explains, the operations of Figures 3 and 4 are executed by the microprocessor unit 44 in the receiver of the GDO system

after that system has been placed in its program mode [see the "IN PROGRAM MODE?" test at the upper right of Figure 3] and the receiver has received an identity code to be programmed or learned, and after the microprocessor unit 44 has executed a software loop to determine whether or not the newly received transmitter identity code has already been stored in the receiver's memory This latter test is used to prevent a given identity code from being stored into more than one memory location.

19. Of note is the fact that the software nature of the loop which causes the microprocessor unit 44 to test to determine whether or not a new transmitter identity code has been received four times in a row (for error-protection purposes) has not been questioned by the Examiner, despite the fact that this loop directly precedes the disclosure which implements that loop for the code location pointer and uses identical flowchart symbols and similar terminology (e.g., "increment").

20. The code location pointer shown in the flow chart of Figure 3 and cited in the specification of the '610 application serves as a moveable identifier which designates (or points to) a specific location in the memory of the GDO system's receiver. As the software loop disclosed in Figure 3 is executed by the microprocessor unit 44, the value of the code location pointer will be moved from one address to the next until all five addresses in memory that were set aside

for transmitter identity codes have been used to store learned identity codes. The specific disclosure of this loop is (a) "Increment Code Location Pointer", followed by (b) "If Pointer Increments Over Five Then Load Code Location Pointer With One." Step (a) is the "body" of the loop; it uses the "increment" operational to increase the value of the code location pointer by one. Step (b) is the test used to terminate the loop by "loading" the code location pointer with "one," its initial value.

21. The language used in the flow chart to describe the loop which implements the software memory address switch (code location pointer) can only be descriptive of a software-based action. The terms "pointer," "increment," "load, and "if. . then," are all specific to the software art wherein, and only wherein, they each have well understood meanings. For example, the Standard Dictionary of Computers and Information Processing, Revised Second Edition, by Martin H. Weik (Hayden Book Company, Inc.) (1977) at p. 182 (Exhibit H) defines "increment" as "to add a quantity to another quantity; for example, to advance the count or number stored in a counter or register." The 1993 edition of the IEEE Dictionary, (Exhibit I) promulgated the standard definition of "increment" as "to increase the value of variable by one." The IEEE Dictionary (1984) (Exhibit B) defines "load" as "to place data into internal storage," linking this definition to another term of the software art "move." [Of note is the fact that the

specification of the '610 application uses the term "move" to describe the action of the memory selection switch, as at Column 3, Lines 7, "The switch 23 can be moved to a number of selected position indicated by the 1 through 5 so as to allow the particular code of a number of different transmitters to be stored in the receiver so that the receiver will respond to such codes."] And, the "if/then" nature of the disclosed code location pointer loop is typical of many, if not most, programming languages. The popular BASIC language provides exactly this programming structure, and the "C" language provides an "if-else" control statement.

22. For each of the terms from Figure 3 discussed above, that terminology will be readily recognized by those skilled in the art as, and only as, being descriptive of a software based method for moving the software memory selection switch so that it will "point" to a new memory location by (a) maintaining the value of that pointer in a memory location of its own or in a register within the microprocessor of the GDO; (b) increasing the value of that pointer by one each time a new transmitter's identity code is to be programmed into the GDO; (c) testing the incremented value of the pointer to see if it has moved past the end of the memory area designated for storing the programmed transmitter identity codes; and (d) if the end has been passed, loading a new value into the pointer which will reset it to its starting value

LX 1491

23. Despite this clear disclosure, the Examiner on Page 8, Lines 9 to 14 of his May 1, 1998 Office Action, states the opinion that, "There is no indication that the incrementing of the location pointer in Figure 3 is a separate software embodiment and this incrementing is interpreted [by him] as controlled by movement of switch 23 to the next position based on several references to switch 23 determining the memory location in applicant's disclosure " He has made this statement despite the obvious software-based nature of the disclosure made in Figure 3 and Column 4, and in a reversal of his previous agreement that the specification of the '118 patent did disclose a software embodiment.

24 In making the assertion quoted above, the Examiner has properly cited the "several references" made in the specification of the '610 application to the role of the hardware memory selection switch 23 in "determining the memory location" at which each new transmitter identity code is stored. He has completely ignored, however, the disclosure of the alternate software code location pointer which performs that same function as shown in Figure 3 and explained in the specification at Column 4, Line 51 to 57.

25 Further, the Examiner has completely and inappropriately ignored the fact that the language of Figure 3 is in no way representative of a mechanical address-changing method or apparatus. Mechanical switches which are operated manually are said to be "repositioned" or "advanced," not

LX 1492

incremented. The term "increment" would rarely, if ever, be interpreted by one of ordinary skill in the art to describe the manual movement of mechanical switch.

26. Similarly, the returning of a mechanical switch to its original position would never be described as "loading" that switch with a specific numerical value.

27 In addition, a mechanical switch cannot be moved past its maximum position and then tested to determine whether or not it needs to be repositioned. Either such a switch will reach a detent (a mechanical stopping point) when it reaches its maximum position, or if it is circular or rotary in nature it will naturally move from its highest position to its initial position as its movement continues in the same direction. Mechanical switches by their nature are self-limiting. A combined "increment" (counter) and "test" (for branch condition) sequence for their operation would be unnecessary and even ill-advised.

28. Further, if the Examiner is correct that the incrementing, testing, and loading actions described for controlling the code location pointer as shown in Figure are in some way descriptive of a manual repositioning method, it is noteworthy that nowhere in the specification of the '610 application is a description provided of the need for the operator of the GDO to turn switch 23 past its fifth position, to observe that the switch has reached a sixth position, and

LX 1493

then to reposition the switch by rotating it in reverse to position no 1.

29 The flowchart disclosed in Figure 3 is a complete and sufficient disclosure of a software embodiment for control of the address at which each new transmitter identification code is to be stored in the memory of the disclosed invention. The method disclosed in that flowchart will readily be implemented in actual software by one of ordinary skill in the software art In so doing, he or she will use a memory location or a register to hold the disclosed code location pointer, initializing it to the starting address of the memory area designated for storage of the identity code, moving it to point to new addresses by use of the increment instruction common to most computer languages, and using a compare instruction to determine whether or not the incrementing instruction has moved the pointer past the end of the designated storage area. If it has reached that condition, the software will load the code location pointer with the address from which it was started.

30. In accordance with the software instructions disclosed in Figure 3, the code location pointer will normally be represented as a binary value held in either a location or position in the memory connected to the microprocessor, or in a register associated with that microprocessor. As such, its binary values will serve to establish specific electrical states for each of the electronic devices used to implement

such a memory location or register. Thus, as the value of the code location pointer is changed during the repetitive operation of the disclosed loop, the electrical conditions of those devices are switched from one state representing the "zero" logical condition to an alternate state representing the "one" logical condition, or the reverse, collectively thereby representing the address of the allocated memory locations

31. Thus, the changing states of the electronic structural elements serve to move or change the value of the code location pointer in accordance with the software-based identity code programming loop disclosed in Figure 3

32 The body of the disclosed software loop shows the steps of:

> (a) Storing the new code in the location of the memory currently pointed to, selected or addressed by the code location pointer (Note that the code is "new" because of the test made immediately before entering the loop which assures that the received code is not already stored in the memory of the receiver)
>
> (b) Moving the code location pointer to the next higher position through the use of an "increment" instruction that causes an increment operation to occur The act of incrementing causes the pointer to move, one by one, through a sequence of addresses. (Note that when incrementation of a pointer is used to designate storage locations, those locations are addressed as contiguous memory locations, e.g., as 5000, 5001, 5002, 5003, and 5004. The locations cannot be scattered throughout the memory address space.) In the illustrative embodiment of Figure 3 this step moves the code location pointer from 1 to 6. (Note that incrementation takes place after the step of storing the new code, thereby moving the code location pointer to the value of "6" after a code has been stored in location "5.")

(c) Testing to determine if the act of incrementing has moved the code location pointer past the last address in memory for storing a code related to one of the five transmitters and, if it has, looping the code location pointer back to point to the first memory location of the five by restoring the value held in the code location pointer to its initial value of "1." Thereafter, the learning of a new code will overwrite the code currently stored in location "1." The test for a code already being anywhere in the set of five memory locations, however, will prevent the same code from occupying more than one of the five memory code locations.

33. The software program described in Paragraph 32 above can readily be performed on any of the several microprocessors available at the time the original application was filed, including the microprocessor disclosed in U.S. Patent 4,638,433 to Schindler. That patent, which was specifically referenced stating that the '610 application comprises an improvement over it, disclosed the use of the microprocessor to control the learning of a single transmitter identity code

34. The "RAM register bank" of columns 20, 21 and 22 of the '433 patent contains a single three-byte memory area used to store the nine-bit manually assigned identity code of the lone radio transmitter of that patent. Those locations are assigned the location name "TRIDAT" (see the "LOCATION" table at the bottom of column 20), corresponding to the trinary encoding which is used in the GDO system of the '433 patent for transmitting data from the lone radio transmitter of that patent to the receiver of the controller. Two software location pointers are disclosed in the '433 patent which show that the '610 application was referring to software pointers,

and not a mechanical switch, when it used the term "code location pointer" in Figure 3.

35 A first software pointer (KEYPT) disclosed in the '433 patent is used to address the four separate memory locations which hold the four-digit security code defined by the user for manual activation of the garage door operation mechanism through the use of a keyboard.

36. A second software pointer (WORDPT) is used only during the transfer of data between the volatile RAM of that patent and the non-volatile EEROM as a means of preserving data should the power supply to the RAM be interrupted (See column 19, lines 38-40, '433 patent).

37. It is therefore my opinion that, in light of the disclosure in Figure 3 of the drawings of the '610 application, the specification of that application, and the disclosure of the Schindler patent, that one of ordinary skill in the art applicable to the '610 application would be impelled to conclude that, in addition to the disclosure of a hardware method and apparatus for controlling the addressing process used to learn multiple transmitter identity codes, the application provides an enabling disclosure of a software method for implementing that process.

All statements made of my own knowledge are true, and all statements made on information and belief are believed to be true. I have been warned that willful, false statements and

the like are punishable by fine or imprisonment, or both (18 U S.C. § 1001) and may jeopardize the validity of the application or of any patent issuing thereon.

V. Thomas Rhyne, Ph.D., P.E.

LX 1498

PATENT

REISSUE LITIGATION

ATTORNEY DOCKET NO. 59838

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicants:	Carl Heitschel Colin Willmott Wayne Schindler	
Appln. No.:	08/700,610	
Filed:	August 12, 1996	
Title:	CODING SYSTEM FOR MULTIPLE TRANSMITTERS AND A SINGLE RECEIVER FOR A GARAGE DOOR OPENER	
Group Art Unit:	2735	
Examiner:	Edwin C. Holloway III	

Received
JUN 25 1998
Group 2700

2700

AMENDMENT AFTER FINAL REJECTION

Assistant Commissioner for Patents
Washington, D.C. 20231

Sir:

06/29/1998 STHORNTO 00000167 08700610
01 FC:197 130.00 OP

In response to the Office Action of May 5, 1998, please amend the aforementioned application as follows:

IN THE SPECIFICATION.

Void date: 06/29/1998 STHORNTO
08700610 STHORNTO 00000167 08700610
01 FC:197 130.00 OP

Please correct the paragraphs at column 4, lines 31 and 57 to read as follows:

06/29/1998 STHORNTO 00000168 08700610
01 FC:199 150.00 OP

With reference to the flow diagrams of FIGS. 3 and 4 it be assumed initially that the switch 22 is in the operate position an incoming signal will be

supplied to terminal A in FIG. 3 and an output will be supplied to terminal B which indicates that the switch 22 is not in the program mode but in the operate mode. Terminal B is illustrated in FIG. 4 and the microprocessor compares the incoming code with any codes in the five code locations stored in the memory address storage 47. If these codes match then the error counter is cleared and all other valid counters. If the valid counter receives the code four times [than] then output is supplied to the terminal C which operates the garage door operator. If the valid counter for the code equals less than 4, then the valid code counter is incremented until the valid code counter does equal 4 which actuates the proper output. Relative to FIG. 4 if the input code does not match any of the five stored codes, then the error counter is incremented and when the error counter equals 3 the error counter is cleared and all valid counters are cleared.

If the switch 22 is in the program mode as shown in FIG. 3 when the incoming signal from a transmitter is received, the flow diagram is followed so as to store the new incoming program in the code location pointed to by the code location pointer [23]. It is to be noted that up to five addresses can be stored in the system of the invention.

IN THE CLAIMS:

---

25. ~~37.~~ (Amended) A garage door operator for a garage door in accordance with claim ~~36~~ 23 wherein if a derived code is stored in all the available storage locations, the memory selection means will select one of such storage locations and the processor means causes the derived code to be stored in that location.

LX 1564

REMARKS

Reconsideration and further examination of the application is respectfully requested.

I. THE CONSENT OF ASSIGNEE IS NOW OF RECORD

The Examiner objected to the application under 37 C.F.R. §1.172(a) on the basis that the assignee had not established its ownership interest (Office Action ¶6). Enclosed is a Consent of Assignee executed by an officer of The Chamberlain Group, Inc., the assignee of the instant application. The Consent specifies the reel and frame number where evidence of chain of title is recorded and is signed by Mark B. Tone, an officer of The Chamberlain Group, Inc. It is respectfully submitted that the Examiner's objection as to the lack of consent of assignee is overcome.

II. THE REISSUE DECLARATION IS NOT DEFECTIVE FOR LACK OF CONSENT OF ASSIGNEE

The Examiner rejected claims 20 through 45 as based upon a defective reissue declaration under 35 U.S.C. §251 due to lack of a consent of assignee. It is respectfully submitted that with the enclosed Consent of Assignee, that rejection (Office Action ¶7) is overcome.

III. THE SUPPLEMENTAL REISSUE DECLARATION ACKNOWLEDGES THE DUTY OF DISCLOSURE FOR CONTINUATION-IN-PART APPLICATIONS

The Examiner has taken the position (Office Action ¶9) that the oath or declaration was defective because it did not recite an acknowledgment of the duty of disclosure related to information which occurred between the filing date of the prior application and a national or PCT international filing date of the continuation-in-part application. An executed declaration with the acknowledgement included therein is being executed by

the inventors and will be submitted in the near future. It is respectfully suggested that the Examiner's objection will be overcome by such submission.

IV. FORMAL DRAWINGS HAVE BEEN FILED

The Examiner requested (Office Action ¶10) that the formal drawings be submitted in the instant application. They are submitted herewith. It is respectfully suggested that this ground for rejection is overcome.

V. THE SPECIFICATION HAS BEEN AMENDED TO CONFORM IT TO ALL CERTIFICATES OF CORRECTION THAT HAVE PREVIOUSLY BEEN ENTERED

The Examiner reminded the assignee that it would be appropriate to incorporate prior Certificate of Correction changes into the instant reissue application. In his comments the Examiner recognizes that such changes were introduced into the March 27, 1998 amendment but continues that all such changes should now be entered.

By the present amendment, the paragraph at col. 4 lines 31-50 has been amended to change "than" on line 42 to "then". Further, the numeral "23" at col. 4 line 55 is again corrected. These changes are believed to be all the entered Certificate of Correction changes to Patent No. Re. 35,364 and U.S. Patent No. 4,750,118 which are relevant to the present application. The assignee respectfully submits that the mistakes to lines 42 and 55 of column 4 were inadvertent and of a typographical or clerical nature. No new matter is introduced by the corrections.

A. The June 6, 1990 Certificate Of Correction.

On June 6, 1990, a Certificate of Correction was requested for Patent No. 4,750,118, however this was withdrawn June 11, 1991 and no certificate issued from the USPTO.

LX 1566

B. The January 15, 1997 Certificate Of Correction.

A first Certificate of Correction for Patent No. Re. 35,364 was requested January 15, 1997 which changes lines 38, 52 and 54 of column 4 to correct printing errors which appear only in Patent No. Re. 35,364. Such printing errors do not occur in the instant application which is the specification of Patent No. 4,750,118 and therefore no similar correction is needed in that specification. The January 15, 1997 request also proposed changes to col. 4 line 48 (then) and col. 4 line 61 ([23]). The reasons for the deletion of the reference numeral 23 are set forth in the amendment dated March 27, 1998 at page 19. These last named changes are requested in the present amendment.

C. The June 17, 1997 Certificate Of Correction.

The last request for Certificate of Correction of Patent No. Re. 35,364 is dated June 17, 1997 and made changes to claims 1 and 2 which are no longer in the instant application. No similar correction is needed here.

In summary, the amendment of the specification corrects lines 48 and 61 of column 4. The corrections are the only ones needed to correct this application to the content of the prior Patent No. 4,750,118 and Patent No. Re. 35,364 as modified by the certificates of correction.

VI. THE CLAIMS PATENTABILITY POINT OUT AND DISTINCTLY CLAIM THE INVENTION

The Examiner rejected claims 20 through 21, 33, 35 through 39, 41 and 44 under 35 U.S.C. §112, ¶2, as being indefinite for failure to particularly point out and distinctly claim subject matter which applicants regard as the invention (Office Action ¶¶12, 13-15). The Examiner, however, only identified the alleged indefiniteness on Claims 20, 33 and 35. Therefore, the following argument is restricted to those claims.

More specifically, the Examiner contended it is unclear as to whether claim 20 is limited to a second transmitter. What claim 20 sets forth, and sets forth definitely, is only that the combination requires at least one transmitter, but that the receiver be responsive to receive transmission from a second transmitter and act thereon. There is nothing indefinite about this language as it recites a specific characteristic of the receiver which enables it to act upon transmissions from a second transmitter as well as a first transmitter, However it does not limit the combination itself to more than one transmitter. The receiver may have the capability to process signals from a second transmitter producing a second code, but two transmitters need not be positively recited in order for the receiver to structurally have that capability. Accordingly, claim 20 is not indefinite and should not be objected to or rejected on this basis.

The Examiner indicated that the recitation of "can be" in claims 33 and 35 is indefinite. The words "can be" indicate a present capacity in the combination as to the way the memory means is operable in that the code within the memory means in the claim combination is a changeable code. That code may be changed by placing the switch means in the program mode in one of the plurality transmitters being identified which has a code which differs from the code previously stored in the memory means. The "can be" recitation only indicates the conditional nature of the code change based upon the switch means being in the program mode state and the transmitter being energized having the code different than the previously stored code. It is respectfully submitted that this is not an indefinite recitation as one of ordinary skill in the art would know that the capacity alleged is one which is conditional upon those states being present. The same argument applies to the recitations of claim 35. Claims 33 and 35 are definite.

VII. THE SOFTWARE EMBODIMENT IS REASONABLY DISCLOSED AND SUPPORTS THE CLAIM LIMITATION DIRECTED TO THE SOFTWARE EMBODIMENT

A. The Examiner's Rejection Is Not Proper.

Claims 20 through 45 are rejected under 35 U.S.C. §251 as being based upon new matter. The Examiner contends that the specification of the original patent does not support the limitations; a processor having a code location pointer; a processor controlled code location pointer; a memory selection means; software controlled code location pointer; a memory selector; microprocessor incrementing the code location point; and memory selection means selecting one of the memory addresses to be erased. (Office Action ¶14). Except for the last limitation, all of these limitations are supported by the software embodiment which, as shown below, is adequately described in the instant specification which is the same as that in the original patent.

The Examiner rejected claims 20 through 23, 28, 30, 24 through 45 under 35 U.S.C. §112, ¶1, on the basis that they contain subject matter which was not reasonably described in the specification. The assignee believes that the "24 through 45" rejection is a typographical error and it should read "34 through 45" as claims 28 and 30 stand rejected individually. The basis for the rejection is that the original specification did not enable or support the software embodiment but, according to the Examiner, is limited to a mechanical switch embodiment.

The Examiner's interpretation of the specification is strained. He ignores specific recitations in the specification that more than two embodiments are set forth and he does not interpret the specification as a whole. Rather, he breaks the specification into pieces, looks at each of the pieces as an isolated subset, and concludes that because language is not recited *in haec verba* there is no disclosure. The patent should be interpreted as a whole. An applicant disclosing the manner

LX 1569

in which a stored program system is meant to work is not held to an impossible standard. For instance, the functions that a software program would instruct a computer to perform for controlling a machine may be readily apparent from the specification which describes, for instance, scan paths and parameters for full tray scanning, as occurred in Robotic Vision Systems, Inc. v. View Engineering, Inc., 112 F.3d, 1163-1166 (Fed.Cir. 1997). Flow diagrams or source code listings are not a requirement for adequately disclosing the functions of software, Fonar Corp. v. General Electric Co., 107 F.3d, 1543-1549 (Fed.Cir. 1997). However, software is disclosed in the instant application, and it is clear that the Federal Circuit does regard flow diagrams as disclosing functions of software as is implicit in Fonar.

B. Figures 3 And 4 Are Flow Diagrams Which Is The Software Embodiment.

It is error not to consider flow diagram Figs. 3 and 4, particularly Fig. 3, as disclosing a software embodiment wherein a code location pointer is operated within and under the control of the microprocessor and is operated directly upon the memory address unit.

> When, as here, the specification and drawings contain a description of the claimed invention, albeit not *in ipsis verbis* (in the identical words), then the examiner or Board, in order to meet the burden of proof, must provide reasons why one of ordinary skill in the art would not consider the description sufficient. In re Alton, 76 F.3d 1168, 1175 (Fed.Cir. 1996).

Assignee is submitting the enclosed declaration of Dr. V. Thomas Rhyne showing that one of ordinary skill in the art would consider the disclosure of the software embodiment sufficient.

The level of ordinary skill in the art of the instant application is that of an individual who, as of October 1985, had a college degree in electrical or computer engineering and at

least two years of experience in designing microprocessor based control systems, including significant experience with the creation of microprocessor software programs for controlling such systems and with wireless garage door opener (GDO) systems. (Rhyne Decl. ¶7)

Based upon his review of the specification, Dr. Rhyne found the specification to clearly disclose to one skilled in the art the description of two embodiments for storing and locating transmitter identity codes in the memory of the receiver. (Rhyne Decl. ¶8)

As the Examiner has recognized, the first of these embodiments includes a system incorporating a GDO which makes use of a hardware memory selection switch 23 for determining the specific memory location at which each newly received transmitter identity code is stored as it is programmed (or learned) by the GDO. Because the disclosed memory selection switch 23 is of a hardware nature, a manual action by the user is required to move the switch from a first position to a second position, and so forth, as successive transmitter identity codes are programmed into the opener. (Rhyne Decl. ¶9)

As averred to by Dr. Rhyne, the specification also provides a clear, obvious, and readily understandable disclosure of another embodiment of the disclosed GDO system. This second embodiment includes a software memory selection switch, referred to in the specification and drawings of the instant application as the "code location pointer." The actions of this software memory selection switch are fully described in Figures 3 and 4 of the instant application as well as at Column 4, Lines 31 through 68 of the specification. Unlike the hardware version of the memory selection switch, the software memory selection switch "moves" from a first position to a second position, and so forth, automatically, returning to the starting memory position only when all available positions have been used to store programmed transmitter identity codes. No form of user selectability is

provided (or allowed) for the software memory location switch. (Rhyne Decl. ¶10)

Dr. Rhyne avers that Figures 3 and 4 of the instant application describe, in a form familiar to those skill in the art as of October 1985, the software-based control operations of the microprocessor of the GDO system. In the instant application, that form is referred to as a flow diagram, a term and a graphical format that are both well understood by those of skill in the art as describing the operations of a software program. More specifically, a flow diagram or chart describes in a graphical format the steps executed by a stored program machine, e.g., a microprocessor unit. The rectangles in Figures 3 and 4 define specific actions which are performed by the system under software control, and the diamonds define test points at which a branch may be made based on the results of the test defined within the diamond. An example of this type of description is shown in Introduction to Microcomputers & Microprocessors, by Arpad Barna and Dan I. Porat, (John Wiley & Sons, Inc.), (1976) at p. 18. The two lower right blocks of Figure 3 describe the operation of the portion of the disclosed software control program executing in the microprocessor unit 44 which uses the code location pointer as the software memory selection switch to determine the location in the GDO system's memory at which each newly learned transmitter identity code is to be stored. As one of skill in the art would readily recognize, these two blocks define, in typical and well understood flow chart form, a sequence of software operations executed by the microprocessor unit 44 and referred to in the art of computer science as a "loop". A loop is a set of microprocessor operations which is repeated until an end-of-loop condition is reached. (Rhyne Decl. ¶¶14, 15)

References describing the construction and use of software loops exactly as disclosed for controlling the code location pointer were readily available as of October 1985. The

previously cited IEEE Dictionary, for example, defines "loop" as "a sequence of instructions that is executed repeatedly until a terminal condition prevails," and the Leventhal textbook defines "loop" as "a self-contained sequence of instructions that a processor repeats until a terminal condition is reached. A conditional jump instruction can determine if the loop should be continued of terminated." Leventhal, in fact, devotes a twenty-page section of his textbook to the subject of "Loops and Arrays" (Section 5.2, pages 179 to 198). A loop includes a body containing the instructions which are the principal business of the loop. An adjustment step is embodied in a set of instructions that adds to a counter of the number of repetitions already performed. An exit from the loop is provided by way of a conditional branch. The conditional branch involves a test for occurrence of certain conditions, as shown, for example, in Assembler Language Programming: The IBM System 360, by George Struble, (Addison-Wesley Publishing Co.), (1971), at pp. 148-151. (Rhyne Decl. ¶¶16, 17)

C. The Term "Code Location Pointer" Tells One Of Ordinary Skill There Is A Software Embodiment.

The specification and drawings refer to the software memory selection switch as a "code location pointer." This term, in and of itself, clearly designates a software version of the memory selection switch. "Code" refers to the new transmitter identity code that is to be programmed into the GDO. "Location" refers to the specific memory address at which that identity code is to be stored and is a term commonly used to designate such addresses as shown, for example, at column 4 lines 37 to 39 of the specification. The term "pointer" is (and was, as of October 1985) a commonly used and well understood term for a stored value which identifies the memory address of another value. The IEEE Standard Dictionary of Electrical and Electronics Terms, Third Edition, (IEEE), (1984), for example, defines "pointer" as "a

data item that specifies the location of another data item," and the widely adopted textbook, Introduction to Microprocessors: Software, Hardware, Programming, by Lance Leventhal, (Prentice-Hall), (1978), defines "pointer" as "[a] register or memory location that contains an address rather than data." In the case of the "code location pointer", its value as disclosed, serves to identify the location within the memory of the GDO system at which the next transmitter identity code to be learned is to be stored. As averred to by Dr. Rhyne, this is precisely what the term "pointer" connotes in the computer science art. (Rhyne Decl. ¶¶11-13)

D. The Test Routine Shows That The Software Embodiment Is Disclosed.

As the specification explains, the operations of Figures 3 and 4 are executed by the microprocessor unit 44 in the receiver of the GDO system after that system has been placed in its program mode [see the "IN PROGRAM MODE?" test at the upper right of Figure 3] and the receiver has received an identity code to be programmed or learned, and after the microprocessor unit 44 has executed a software loop to determine whether or not the newly received transmitter identity code has already been stored in the receiver's memory. This latter test is used to prevent a given identity code from being stored into more than one memory location. Of note is the fact that the software nature of the loop which causes the microprocessor unit 44 to test to determine whether or not a new transmitter identity code has been received four times in a row (for error-protection purposes) has not been questioned by the Examiner, despite the fact that this loop directly precedes the disclosure which implements that loop for the code location pointer and uses identical flowchart symbols and similar terminology (e.g., "increment"). (Rhyne Decl. ¶¶18, 19)

E. Incrementing Shows That The Software Embodiment Is Disclosed.

The code location pointer shown in the flow chart of Figure 3 and cited in the specification of the '610 application serves as a moveable identifier which designates (or points to) a specific location in the memory of the GDO system's receiver. As the software loop disclosed in Figure 3 is executed by the microprocessor unit 44, the value of the code location pointer will be moved from one address to the next until all five addresses in memory that were set aside for transmitter identity codes have been used to store learned identity codes. The specific disclosure of this loop is (a) "Increment Code Location Pointer", followed by (b) "If Pointer Increments Over Five Then Load Code Location Pointer With One." Step (a) is the "body" of the loop; it uses the "increment" operational to increase the value of the code location pointer by one. Step (b) is the test used to terminate the loop by "loading" the code location pointer with "one," its initial value. (Rhyne Decl. ¶20)

F. Flow Chart Language Shows That The Software Embodiment Is Disclosed.

The language used in the flow chart to describe the loop which implements the software memory address switch (code location pointer) can only be descriptive of a software-based action. The terms "pointer," "increment," "load, and "if... then," are all specific to the software art wherein, and only wherein, they each have well understood meanings. For example, the Standard Dictionary of Computers and Information Processing, Revised Second Edition, by Martin H. Weik (Hayden Book Company, Inc.), (1977), at p. 182, defines "increment" as "to add a quantity to another quantity; for example, to advance the counter or number stored in a counter or register." The IEEE Dictionary, (1993), promulgates the standard definition of "increment" as "to increase the value of variable by one." The IEEE Dictionary

(1984), defines "load" as "to place data into internal storage," linking this definition to another term of the software art "move." [Of note is the fact that the specification uses the term "move" to describe the action of the memory selection switch, as at Column 3, Lines 7, "The switch 23 can be moved to a number of selected position indicated by the 1 through 5 so as to allow the particular code of a number of different transmitters to be stored in the receiver so that the receiver will respond to such codes."] And, the "if/then" nature of the disclosed code location pointer loop is typical of many, if not most, programming languages. The popular BASIC language provides exactly this programming structure, and the "C" language provides an "if-else" control statement. (Rhyne Decl. ¶21)

As averred to by Dr. Rhyne, for each of the terms from Figure 3 discussed above, that terminology will be readily recognized by those skilled in the art as, and only as, being descriptive of a software based method for moving the software memory selection switch so that it will "point" to a new memory location by (a) maintaining the value of that pointer in a memory location of its own or in a register within the microprocessor of the GDO; (b) increasing the value of that pointer by one each time a new transmitter's identity code is to be programmed into the GDO; (c) testing the incremented value of the pointer to see if it has moved past the end of the memory area designated for storing the programmed transmitter identity codes; and (d) if the end has been passed, loading a new value into the pointer which will reset it to its starting value. (Rhyne Decl. ¶23).

Despite this clear disclosure, the Examiner on Page 8 of the Office Action, states his opinion that, "There is no indication that the incrementing of the location pointer in fig. 4 (sic) is a separate software embodiment and this incrementing is interpreted [by him] as controlled by movement of switch 23 to the next position based on several references to switch 23 determining the memory location in applicant's disclosure."

Further, on page 21, the Examiner again asserts that the flow chart of Figure 3 represents "moving the switch 23 by the user to the next pointer". The Examiner makes these statements despite the obvious software-based nature of the disclosure made in Figure 3 and Column 4, and in a reversal of his previous agreement that the specification of the '118 patent did disclose a software embodiment.

In making the assertions quoted above, the Examiner has properly cited the "several references" made in the specification of the instant application to the role of the hardware memory selection switch 23 in "determining the memory location" at which each new transmitter identity code is stored. He has completely ignored, however, the disclosure of the alternate software code location pointer which performs that same function as shown in Figure 3 and explained in the specification at Column 4, Line 51 to 57. (Rhyne Decl. ¶24)

G. The Language Describing The Software Embodiment Of The Invention Is Not Consistent With The Disclosure Of Just A Mechanical Switch Embodiment.

Further, the Examiner has completely and inappropriately ignored the fact that the language of Figure 3 is in no way representative of a mechanical address-changing method or apparatus. Mechanical switches which are operated manually are said to be "repositioned" or "advanced," not incremented. The term "increment" would rarely, if ever, be interpreted by one of ordinary skill in the art to describe the manual movement of mechanical switch. (Rhyne Decl. ¶25)

Similarly, the returning of a mechanical switch to its original position would never be described as "loading" that switch with a specific numerical value. (Rhyne Decl. ¶26)

In addition, a mechanical switch cannot be moved past its maximum position and then tested to determine whether or not it needs to be repositioned. Either such a switch will reach a

detent (a mechanical stopping point) when it reaches its maximum position, or if it is circular or rotary in nature it will naturally move from its highest position to its initial position as its movement continues in the same direction. Mechanical switches by their nature are self-limiting. A combined "increment" (counter) and "test" (for branch condition) sequence for their operation would be unnecessary and even ill-advised. (Rhyne Decl. ¶27)

Further, if the Examiner is correct that the incrementing, testing, and loading actions described for controlling the code location pointer as shown in Figure are in some way descriptive of a manual repositioning method, it is noteworthy that nowhere in the specification of the instant application is a description provided of the need for the operator of the GDO to turn switch 23 past its fifth position, to observe that the switch has reached a sixth position, and then to reposition the switch by rotating it in reverse to position no. 1.

H. Coupling The Incrementing Code Location Pointer With A Receipt Of Code Four Times For Storage Thereof Requires The Software Embodiment.

The flow chart of Figure 3 requires that before the code location pointer, which the Examiner considers to be the switch, is incremented, the code to be stored must be received four times and then the microprocessor unit 44 determines whether the code is already stored in the memory. The user would have no way of knowing whether these two conditions are met and therefore would not know when to move the switch. Thus, the increments of the code locations can only be done by software. The reasoning engaged in by the Examiner in order to avoid holding that software is disclosed in the application is incorrect and constitutes reversible error.

In view of the above, claims 20 through 45 were improperly rejected under 35 U.S.C. §251, as new matter was not added to the

LX 1578

application. Claims 20 through 23, 28, 30 and 34 through 45 were improperly rejected under 35 U.S.C. §112, ¶1, as the subject matter of these claims was adequately described in the specification.

VIII. THE AMENDMENT TO CLAIM 37

In paragraph 15 of the Office Action the Examiner asserts that the original specification does not support "the memory selection means selecting one of the memory addresses to be erased in claim 37" stating that only disclosure of the erase limitation is the description that additional codes 'erase all codes' in col. 4, line 67 which cannot support or enable the limitation of memory selection means selecting one of the 'addresses to be erased.' The Examiner is not reading the disclosure in context. The remainder of the specification discloses that one code is stored in the memory at one selected location. In order for the code to be stored at this location the location must be empty or any code previously stored at that location must be erased. The subject disclosure states

> Each time the "program mode switch" 22 is moved to the program position, a different storage area as determined by the switch 23 can be connected so that the new transmitter code would be stored in that address. After all of the address storage capacity has been used additional codes would erase all old codes in the memory address storage before storing a new one... (col. 4, ll 61-68).

Thus, this means that after all of the memory locations are filled, additional codes would erase all prior stored codes one at a time in response to new codes being stored. However, to reduce the issues, claim 37 has been amended to overcome the Examiner's objection to the limitation "erase all codes."

LX 1579

IX. THE CLAIMS ARE PATENTABLE OVER THE PRIOR ART

The Examiner relies upon three art combinations to reject claims 20-45 under 35 U.S.C. § 103(a). The first obviousness rejection is based on Tsubaki patent, No. 4,385,296 in combination with the Early patent, No. 4,369,491 (Office Action ¶17). The second obviousness rejection is based upon the Pinnow patent, No. 4,665,397 or the Pinnow patent, No. 4,573,046 in view of the Tolson patent, No. 3,337,992 and the Early patent, No. 4,369,481 (Office Action ¶18). The third obviousness rejection is based upon the Pinnow patent No. 4,663,397 or the Pinnow patent No. 4,573,046 in combination with the Tsubaki patent, No. 4,385,296 and the Early patent, No. 4,369,481 (Office Action ¶19).

The Patent and Trademark Office allowed the original patent No. 4,750,118, and the claims of the first reissue patent, U.S. Patent No. Re. 35,364, over the Tsubaki patent. The Pinnow '397 patent is a continuation-in-part of the Pinnow '046 patent and thus, the Pinnow '397 patent contains the disclosure of the Pinnow '046 patent.[1]

The legal conclusion as to whether the subject matter set forth in claims 20-45 is obvious or nonobvious is one which cannot be solved by hindsight considerations. It is clear that in this instance the Examiner has employed impermissible hindsight analysis in order to defeat the patentability of the pending claims. The Examiner has decomposed the invention into its constituent elements, found each element in the prior art, and then claimed that it is obvious to reassemble these elements into the claimed invention. This is a forbidden *ex post* analysis. It is impermissible simply to engage in a hindsight reconstruction of the claimed invention, using the applicant's

---

[1] Because the Pinnow '397 patent contains the complete disclosure, we will only cite to the specification of the Pinnow '397 patent.

structure as a template in selecting elements from references to fill the gaps. In re Gorman, 933 F.2d 982, 987 (Fed.Cir. 1991); In re Pleuddemann, 910 F.2d 823, 827 (Fed.Cir. 1990)(it is error to use applicant's specification teaching as though it were prior art in order to make claims appear to be obvious).

A. The Early Patent, The Pinnow Patent And The Tolson Patent Are Non-Analogous Art.

In order to rely on a reference as a basis for rejection of the claimed invention, the reference must either be in the field of the inventors' endeavor or, if not, then the reference must be reasonably pertinent to the particular problem with which the inventors were concerned. See In re Oetiker, 977 F.2d 1443, 1447 (Fed.Cir. 1992).

1. Early Is Not Pertinent To Applicants' Problem Because Early Does Not Describe Code Or Electrical Signals Stored In Memory, But Rather A Key Which Physically Interacts With A Lock Via Optical Patterns Which When Matched Permit The Unlocking Of The Lock.

In his rejections using Early in the above three art combinations, the Examiner mainly relies upon the second embodiment of the electronic lock disclosed in the Early patent (see Fig. 4). The Early patent, which is in the electronic lock art, must have its mechanical key physically inserted into the lock mounted at the door to open the door, in the path of a laser beam in order that the light reflecting pattern on the key may be scanned by the laser beam. The key must be in the lock to provide a circuit path to power up the lock (Col. 3, lines 39-41). The light reflecting means may be a crystal, a plurality of small mirrors positioned to reflect light in the unique pattern, or made of many small light reflecting flakes embedded in plastic (see col. 2, lines 59-66). This is not the same thing as a code, which is an electrical signal, stored in a memory. Because of the difficulty and expense needed to arrange a

crystal, mirror or flakes in numerous patterns which must be matched by the series-connected photo-diodes, the Early patent clearly would not be pertinent to the inventors' problem of providing a very large number of uniquely coded transmitters

In the second disclosed embodiment, the key must be inserted in the lock to turn it on. The microprocessor then waits for the person to punch in a number identifying the key. (Col. 3, lines 39-47). Photodiodes D1 through D20 are located in an array in the lock which is scanned to determine if the laser that is reflected from the reflective means on the inserted key matches a programmed pattern in a memory. The memory is programmed with new key codes by scanning new keys when they are in the lock and storing the codes corresponding to the reflected light pattern, but not the pattern itself.

Despite the Examiner's contention that the Early patent discloses "an analogous art microprocessor controlled door access control system," (Office Action, p. 9) it is a non-analogous optical system. The Early patent is not in the field of remote-controlled openers for garage doors, and is not reasonably pertinent to the problems with which the inventors were concerned, i.e., eliminating code changing switches in the transmitters and receivers of garage door openers while obtaining a substantial number of available codes so as to provide a secure system. (Willmott Decl. ¶¶11,12) A person skilled in that art would not look to the electronic lock art where a mechanical key is inserted into the lock mounted at the door to open the door.

The Early patent is not analogous art also because the reflecting means of the Early patent only provides a very limited number of available patterns. These are not fixed codes as claimed. Also, one skilled in the art would not look to the Early patent because a laser beam system in which the user must insert a mechanical key into a lock to be read by the laser beam and diodes is not practical for use in the operation of a garage door opener, which requires remote operation by a radio frequency

transmitter. The user of the Early system would need to get out of the car to insert the key into the sensor to operate the garage door opener. This would be a huge step backwards in the technology. In addition, because the sensor of the Early patent, if used in a garage door opener system, would have to be mounted outside the garage, a communication channel would need to be provided, and a laser beam because of stringent columnation demands would not be practical. There is no teaching or suggestion in Early of other than a laser beam system which must have its elements precisely placed in order to operate. Accordingly, the Early patent is not analogous art. It is not within the content of the prior art and can not be relied upon as a reference.

2. Pinnow's Optical Signal Transmitting Device Is Not Relevant To Assignee's RF System Which Requires A Receiver With Fixed Code.

The Pinnow patents can not be relied upon as prior art because they are not analogous prior art. The Pinnow patents are not in the field of the inventors' endeavor and not necessarily pertinent to the problem with which the inventors were concerned. The Pinnow's optical signal-transmitting unit contained in a watch for operating within a very short distance from an electronic locking system. This type of unit would not commercially function out of doors as would be required for a GDO unit as claimed.

The Pinnow patents disclose that during normal operation the transmitter unit is not operated until the user is within arm's length of the lock. The infrared beam emitted by the watch must be aimed at the lock by line-of-sight with the eye and aimed blindly as the beam is invisible. (Pinnow '397 patent, col. 4, lines 20-24). Such critical transmitter alignment would be almost impossible in those cases when the transmitter is not in line of sight with the receiver due to the limited diffraction of the infrared beam around intervening obstacles. This is not

in the field of the invention of remote controlled openers for garage doors.

Even though Pinnow makes a passing reference to the use of a fiber optic relay for opening a garage door, he specifically refers to actuating a lock mechanism on the garage door (Pinnow '397 patent, Col. 9, lines 58-61). The lock mechanism disclosed by Pinnow is a signal receiving unit and latching mechanism. (Col. 7, lines 8-40) That mechanism, however, does not teach a garage door opener wherein a motor is actuated to move the door up or down. Even the emitting end of the fiber optic relay still requires the arm's length and line-of-sight actuation which would make the Pinnow patent impractical for a garage door operator. (Willmott Decl. ¶14)

Also, the Pinnow patents are classified in 340/825.56; 340/825.72 and 361/172. The subject application is classified in 340/825.69. The Patent No. Re 35,364 is classified in 364/400; 49/25; 340/825.69 and 340/825.73. These disparate classifications by the PTO further evidence that the Pinnow patents are non-analogous prior art. A person skilled in the art would not consider a fiber optic relay operated by an infrared light beam emitted by a watch to solve the problem of DIP switch miscoding and lack of security in radio frequency transmitters. (Willmott Decl. ¶14) Thus, the Examiner can not rely upon the Pinnow patents as prior art.

3. Tolson Is Non-Analogous Art Because It Does Not Describe Or Suggest The Transmission Of Code By RF.

The Tolson patent also is not an analogous art. It is directed to closures for windows or doors that are controlled by two or three position switches or two position push buttons. The transmission of codes by a radio frequency signal is not disclosed. Therefore, the patent is not in the field of the inventors' endeavor (i.e., radio frequency operated garage door openers), and would not be reasonably pertinent to the particular

problem with which the inventors were concerned (i.e., eliminating DIP switches and providing security in transmission of codes). (Willmott Decl. ¶13)

B. The Combined Prior Art Patents Do Not Teach Or Suggest All The Claim Limitations.

1. None Of The References Teach Or Suggest A Plurality Of Radio Frequency Transmitters Having Non-User-Changeable Codes That Are Different From Each Other.

To establish *prima facie* obviousness of a claimed invention, all the claim limitations must be taught or suggested by the prior art. MPEP, § 2143.03. None of the patents relied upon by the Examiner teaches or suggests a plurality of radio frequency transmitters having non-user-changeable codes that are different from each other, as called for in the claims. Hence, they combination cannot be combined to suggest this important feature of the claimed invention. The Tsubaki patent, as admitted by the Examiner on page 9 of the Office Action, discloses a single transmitter which has user-changeable DIP switches for changing the code (see col. 3, lines 27-39). In the non-analogous Pinnow patents, the code stored in the memory of the watch for use by a lock receiver is user-changeable. The user selects whatever code the user desires by depressing the code key 24 for a selected length of time for each digit of the code (see Pinnow Pat. '397, col. 8, lines 26-54).

Contrary to the Examiner's comments on page 12 of the Office Action, the Pinnow '397 patent does not disclose "fixed code transmitters" for a single receiver in its description of the prior art. Also, the Pinnow patents disclose replacing a typical key with a user-changeable code transmitter, not with a non-user-changeable code transmitter, as asserted by the Examiner (Office Action, p. 13).

There should be no doubt that the non-analogous Tolson patent does not disclose transmitters with different, non-user-changeable codes, and the Examiner does not assert that it does.

The non-analogous Early patent does not disclose a plurality of transmitters much less radio frequency transmitters having different, non-user-changeable codes. The mechanical keys of Early are not radio frequency transmitters or any other kind of transmitter and the pattern of the light reflecting means is not a code.

2. Early's System Of Optically Reflecting Patterns As Part Of A Combination With The Other References Does Not Suggest Non-User Changeable Codes.

Realizing that the cited prior art does not teach the claimed plurality of transmitters with non-user-changeable codes, the Examiner contending that because Early's disclosure of a plurality of mechanical keys which have thereupon non-user-changeable reflecting patterns, suggests plural radio frequency transmitters with non-user changeable codes in the Pinnow patent or the Tsubaki patent. This is specious. Mechanical patterns have no relation to electrical codes.

Tsubaki also does not suggest such transmitters. Tsubaki discloses replacing DIP switches in the receiver but this does not suggest replacing DIP switches in the transmitter because Tsubaki did not do so.

Thus, because none of the prior art in the Examiner's three asserted combinations shows or suggests this claimed feature of a plurality of radio frequency transmitter with non-user-changeable codes, the claims are patentable for this reason alone. The three combinations of references relied upon by the Examiner fail to render the subject matter of the claims obvious because the combined references do not teach or suggest all of the claim limitations. This claimed feature of non-user-changeable codes provides the garage door opener with many